ระบบตรวจจับความชื้นในผ้าอ้อม

SMART DIAPER

โดย

นางสาวปานหทัย	แซ่เตี๋ย	รหัส	B5304776
นางสาวณัฐฐา	คลังสินธ์	รหัส	B5312009
นางสาวกีรติกานต์	ใจเอื้อ	รหัส	B5313396

รายงานนี้เป็นส่วนหนึ่งของการศึกษาวิชา 427499 โครงงานวิศวกรรมโทรคมนาคม
หลักสูตรวิศวกรรมศาสตร์บัณฑิต สาขาวิชาวิศวกรรมโทรคมนาคม หลักสูตรปรับปรุง พ.ศ. 2546
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี
ประจำภาคการศึกษาที่ 2 ปีการศึกษา 2556

ระบบตรวจตรวจจับความชื้นในผ้าอ้อม

คณะกรรมการสอบโครงงาน

(ผู้ช่วยศาสตราจารย์ ดร. วิภาวี หัตถกรรม)
กรรมการ/อาจารย์ที่ปรึกษาโครงงาน

(ผู้ช่วยศาสตราจารย์ ดร. ชาญชัย ทองโสภา)
กรรมการ

(อาจารย์ ธนเสฏฐ์ ทศดีกรพัฒน์)
กรรมการ

มหาวิทยาลัยเทคโนโลยีสุรนารี

มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้นับรายงานโครงงานฉบับนี้ เป็นส่วนหนึ่งของการศึกษาระดับปริญญาตรี สาขาวิชาวิศวกรรมโทรคมนาคม รายวิชา 427499 โครงงานวิศวกรรมโทรคมนาคม ประจำปีการศึกษา 2556

โครงงาน	ระบบตรวจจับความชื้นในผ้าอ้อม (Smart Diaper)		
ผู้จัดทำ	1. นางสาวปานหทัย	แซ่เตี๋ย	รหัส B5304776
	2. นางสาวณัฐฐา	คลังสินธ์	รหัส B5312009
	3. นางสาวกีรติกานต์	ใจเอื้อ	รหัส B5313396
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร.วิภาวี หัตถกรรม		
สำนักวิชา	วิศวกรรมศาสตร์		
สาขาวิชา	วิศวกรรมโทรคมนาคม		
ภาคการศึกษา	2/2556		

บทคัดย่อ

โครงงานนี้เป็นการทำระบบตรวจจับความชื้นในผ้าอ้อมสำหรับ เด็กทารก ผู้สูงอายุ และผู้ป่วยที่ไม่สามารถช่วยเหลือตัวเองได้ โดยการนำเอาอุปกรณ์สื่อสารแบบไร้สาย(Xbee) มาประยุกต์ใช้ทรานสมิตเตอร์ในการทำเซ็นเซอร์เหนี่ยวนำความชื้นอย่างง่าย เพื่อตรวจจับความชื้นที่เกิดขึ้นบนผ้าอ้อม โดยเหนี่ยวนำให้เกิดความต่างศักย์ที่เปลี่ยนไป ส่งไปยังบอร์ดไมโครคอนโทรลเลอร์Arduino ที่ต่อเข้ากับอุปกรณ์สื่อสารไร้สายภาคส่งทำการส่งไปยังอุปกรณ์สื่อสารไร้สายภาครับแสดงผลบนจอมอนิเตอร์ ผู้ดูแลได้ทราบว่าผ้าอ้อมชิ้นนั้นๆชื้อและสามารถแจ้งเตือนพยาบาล หรือผู้ดูแลให้เปลี่ยนผ้าอ้อมชิ้นใหม่

มหาวิทยาลัยเทคโนโลยีสุรนารี

กิตติกรรมประกาศ

โครงงานเรื่อง ระบบตรวจจับความชื้นในผ้าอ้อม (Smart Diaper)นี้ สำเร็จลุล่วงไปด้วยดี เนื่องจากได้รับความอนุเคราะห์และคำแนะนำจากบุคคลดังต่อไปนี้

ขอขอบคุณ ผู้ช่วยศาสตราจารย์ ดร.วิภาวี หัตถกรรม อาจารย์ที่ปรึกษาโครงงาน ที่ได้ให้คำชี้แนะ แนวคิด ตลอดจนแก้ไขข้อบกพร่องต่างๆ

ขอขอบคุณ นายปัญญา หันตุลา ที่ได้ให้คำปรึกษาและคำแนะนำเกี่ยวกับการทำเซ็นเซอร์ การใช้งานอุปกรณ์ การใช้งานโปรแกรมต่างๆ

ขอขอบคุณคุณพ่อคุณแม่ เพื่อนๆที่ให้การสนับสนุน คำปรึกษา คำแนะนำ และให้การช่วยเหลือทุกๆด้านด้วยดีเสมอมา

คณะผู้จัดทำโครงงานใคร่ขอขอบพระคุณทุกๆ ท่านที่ได้กล่าวไปแล้วไว้ ณ ที่นี้สำหรับส่วนดีของโครงงานชิ้นนี้ขออุทิศให้แก่คณาจารย์ทุกท่านที่ได้ประสิทธิ์ประสาทวิชาความรู้ให้แก่คณะผู้จัดทำโครงงานหากโครงงานชิ้นดีมีข้อผิดพลาดประการใดทางคณะผู้จัดทำโครงงานใคร่ขอน้อมรับและขออภัยมา ณ ที่นี้

นางสาวปานหทัย แซ่เตี๋ย

นางสาวณัฐฐา คลังสินธ์

นางสาวกีรติกานต์ ใจเอื้อ

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญ

เรื่อง	หน้า
บทคัดย่อ	ก
กิตติกรรมประกาศ	ข
สารบัญ	ค
สารบัญ(ต่อ)	ง
บทที่ 1 บทนำ	**1**
1.1 หลักการ และเหตุผล	1
1.2 วัตถุประสงค์	2
1.3 ขอบเขตของการศึกษาโครงงาน	2
1.4 ผลที่คาดว่าจะได้รับ	2
1.5 ขั้นตอนการดำเนินงาน	3
บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง	**4**
2.1 บทนำ	4
2.2. การใช้ความชื้นเหนี่ยวนำให้เกิดกระแสไฟฟ้าไหลในวงจรผ่านทรานซิสเตอร์	4
2.3.การรับส่งข้อมูลผ่านคลื่นความถี่ในอากาศ	4
2.4.รูปแบบการเชื่อมต่ออุปกรณ์ (ไร้สาย)	5
บทที่ 3 การออกแบบและดารทำงานของระบบ	**6**
3.1 บทนำ	6
3.2 เซ็นเซอร์ความชื้น	6
3.2 การเชื่อมต่อเซ็นเซอร์ความชื้นกับอุปกรณ์สื่อสารไร้สาย(Xbee)และคอมพิวเตอร์	7
3.3 การเขียนโปรแกรมควบคุมบอร์ดไมโครคอนโทรลเลอร์(Arduino)โดยใช้ภาษาซี	13
3.4 การออกแบบโปรแกรมแสดง	19
บทที่ 4 ผลการทดลอง	**27**
4.1 บทนำ	27
4.2 ผลการทดลอง	27
4.3.วิเคราะห์ผลการทดลอง	30

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญ(ต่อ)

เรื่อง	หน้า
บทที่ 5 สรุปผลการทดลอง	**31**
5.1 สรุปผลการทดลอง	31
5.2 สิ่งที่ได้รับจากการทำโครงงาน	31
5.3 ปัญหาและอุปสรรค	31
5.4 ข้อเสนอแนะ	31
เอกสารอ้างอิง	**32**
ภาคผนวก	**33**
ภาคผนวก ก	33
ภาคผนวก ข	40
ประวัติผู้เขียน	65

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 1
บทนำ

การที่นางพยาบาลไม่ต้องเดินถามตามเตียงผู้ป่วยว่าต้องการเปลี่ยนผ้าอ้อมหรือไม่ ซึ่งบางขณะผู้ป่วยบางคนไม่สามารถสื่อสารได้ หรือในกรณีเด็กเล็กที่ยังต้องใช้ผ้าอ้อม จอมอนิเตอร์จะแสดงสัญลักษณ์ให้ทราบว่าผู้ป่วยคนไหนต้องการเปลี่ยนผ้าอ้อมใหม่ เด็กเล็กต้องการเปลี่ยนผ้าอ้อม โครงงานนี้นำเสนอระบบตรวจจับความชื้นในผ้าอ้อม โดยการใช้ทรานสมิตเตอร์()ในการทำเซ็นเซอร์เหนี่ยวนำความชื้นอย่างง่าย โดยใช้ประโยชน์จากความต่างศักย์ที่เปลี่ยนไปในผ้าอ้อมเป็นตัวเหนี่ยวนำให้เกิดการส่งสัญญาณไปยังอุปกรณ์สื่อสารไร้สาร (Xbee) ซึ่งใช้เป็นตัวกลางในการติดต่อระหว่างผ้าอ้อมกับจอมอนิเตอร์แสดงผล โดยการควบคุมระบบการทำงานโดยใช้ภาษาซีบนบอร์ดไมโครคอลโทรลเลอร์ (Arduino) ระบบตรวจจับความชื้นเป็นอีกเทคโนโลยีประยุกต์ใช้เพื่อความสะดวกสบายของมนุษย์ การทำระบบตรวจจับความชื้นในผ้าอ้อมขึ้นเพื่อช่วยลดขั้นตอนต่างๆ ในการดำรงชีวิต

1. 1 หลักการ และเหตุผล

เนื่องด้วยในปัจจุบันเทคโนโลยีได้ก้าวเข้ามาเป็นส่วนหนึ่งในชีวิตประจำวันของมนุษย์ในวัตถุประสงค์ในด้านความสะดวกสบาย ลดขั้นตอนต่างๆในกาดำรงชีวิต ระบบตรวจจับวามชื้นในผ้าอ้อมก็เป็นอีกตัวอย่างหนึ่ง จากการนำเทคโนโลยีมาประยุกต์ใช้เพื่อความสะดวกสบายของมนุษย์ การที่นางพยาบาลไม่ต้องคอยตรวจสอบผ้าอ้อมตามเตียงผู้ป่วยว่าต้องการเปลี่ยนผ้าอ้อมหรือไม่ ซึ่งบางขณะผู้ป่วยบางคนไม่สามารถสื่อสารหรือช่วยเหลือตัวเองได้ ก็ต้องใช้การเคาสุ่มหรือการแตะสัมผัส ซึ่งเสียเวลาและไม่สะดวกแก่ทั้งผู้ป่วยหรือตัวนางพยาบาลคนนั้นเอง หรือในกรณีเด็กเล็กที่ยังต้องใช้ผ้าอ้อม คุณพ่อคุณแม่คงปฏิเสธไม่ได้ว่าต้องคอยตรวจสอบตลอดเวลาว่าลูกน้อยของตนต้องการเปลี่ยนผ้าอ้อมหรือยัง ระบบตรวจสอบความชื้นในผ้าอ้อมสามารถช่วยแก้ปัญหาเหล่านั้นได้ คุณพ่อคุณแม่ นางพยาบาล หรือผู้ดูแลเพียงคอยมองสัญญาณเตือนจากจอมอนิเตอร์ในห้องทำงาน ไม่ต้องเสียเวลาคอยตรวจตรา คุณพ่อคุณแม่สามารถปล่อยให้ลูกน้อยสนุกกับการเรียนรู้ได้อย่างเต็มที่ โดยไม่ต้องกังวลว่าลูกน้อยของตนต้องการเปลี่ยนผ้าอ้อมหรือยัง จอมอนิเตอร์จะบอกเองว่าลูกน้อยของคุณต้องการเปลี่ยนผ้าอ้อมแล้ว สามารถนั่งทำงานได้พร้อมทั้งดูแลลูกน้อยได้ในขณะเดียวกัน

อุปกรณ์เซ็นเซอร์จะตรวจจับความชื้นในผ้าอ้อมส่งไปยังจอแสดงผลโดยการใช้อุปกรณ์สื่อสารไร้สาย (Xbee) ในการติดต่อระหว่างผ้าอ้อมกับจอแสดงผล โดยการควบคุมระบบการทำงานโดยใช้บอร์ด Arduino

1.2 วัตถุประสงค์

1. เพื่อศึกษาขั้นตอนและวิธีการใช้งานของอุปกรณ์สื่อสารไร้สาย(Xbee)
2. เพื่อศึกษาขั้นตอนและวิธีการใช้งานของบอร์ด Arduino
3. สามารถใช้โปรแกรมภาษา C ในการคอนโทรลบอร์ด Arduino และเซ็นเซอร์ได้
4. สามารถนำความรู้จากทฤษฎีมาประยุกต์ใช้ในระบบตรวจจับความชื้นในผ้าอ้อมได้จริง
5. สามารถทำงานเป็นทีม และแก้ปัญหาที่เกิดขึ้นได้

1.3 ขอบเขตของการศึกษาโครงงาน

1. ศึกษาวิธีการใช้งานอุปกรณ์สื่อสารไร้สาย(Xbee)
2. ศึกษาวิธีการใช้งาน Arduino และโปรแกรมที่ใช้สั่งการ
3. ศึกษาการเชื่อมโยงระหว่างเซ็นเซอร์ บอร์ดArduino อุปกรณ์สื่อสารไร้สายภาครับ/ภาคส่ง และจอมอนิเตอร์เข้าด้วยกัน
4. จอมอนิเตอร์สามารถแสดงผลได้อย่างถูกต้อง

1.4 ผลที่คาดว่าจะได้รับ

1. สามารถสร้างอุปกรณ์ต้นแบบสำหรับตรวจจับความชื้นในผ้าอ้อมแก่ เด็กทารก ผู้สูงอายุ และผู้ป่วยที่ช่วยเหลือตัวเองไม่ได้
2. สามารถนำความรู้ที่ได้มาใช้ในการประกอบวิชาชีพ
3. สามารถนำความรู้ทางทฤษฎีมาประยุกต์ใช้ในทางปฏิบัติ
4. มีความรู้ความเข้าใจในการใช้งานโปรแกรมภาษาซี
5. สามารถทำงานเป็นทีมได้

มหาวิทยาลัยเทคโนโลยีสุรนารี

1.5 ขั้นตอนการดำเนินงาน

1. ศึกษา ค้นคว้าหาข้อมูล
2. เขียนโครงการและเสนอโครงการกับอาจารย์ที่ปรึกษา
3. หาซื้ออุปกรณ์ที่ใช้ในโครงการนี้
4. ออกแบบกลไกการทำงานของระบบโดยรวมทั้งหมด
5. ออกแบบวงจรการเชื่อมต่อเพื่อใช้งานของแต่ละอุปกรณ์
6. เขียนโปรแกรมควบคุมการทำงานบอร์ด Arduino
7. เขียนโปรแกรมเชื่อมต่ออุปกรณ์สื่อสารไร้สายภาครับ/ภาคส่ง
8. เชื่อมต่ออุปกรณ์ทั้งระบบเข้าด้วยกัน
9. จำลองการทำงานของอุปกรณ์ต้นแบบและทดสอบเพื่อให้ได้ตามวัตถุประสงค์
10. สรุปผลการทดลองและเขียนรายงาน
11. นำเสนอโครงงาน

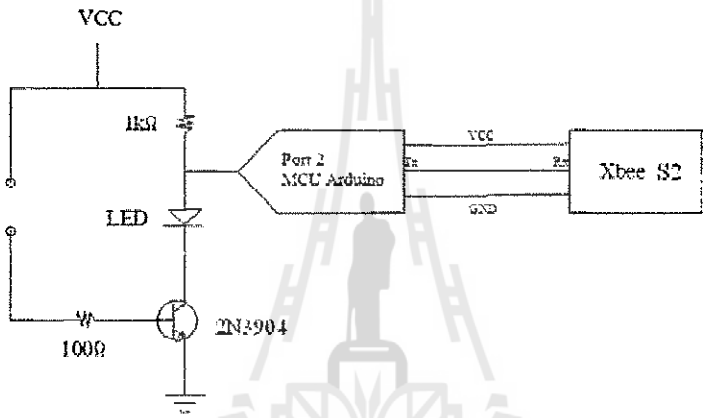
มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 2
ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 บทนำ

ในบทนี้จะกล่าวถึงทฤษฎีและหลักการของการใช้ความชื้นเหนี่ยวนำให้เกิดกระแสไฟฟ้าไหลในวงจรผ่านทรานซิสเตอร์ การรับส่งข้อมูลผ่านคลื่นความถี่ในอากาศ

2.2. การใช้ความชื้นเหนี่ยวนำให้เกิดกระแสไฟฟ้าไหลในวงจรผ่านทรานซิสเตอร์



ความชื้นที่ผ้าอ้อมจะทำให้มีกระแสไหลครบวงจร จากนั้นกระแสจะไหลผ่านอุปกรณ์ต่างๆ แล้วเข้าสู่ ขาที่ 2 ของบอร์ด Arduino เพื่อสั่งงานให้ Xbee ส่งค่าผ่านสายอากาศไปยัง Xbee ตัวรับที่อยู่กับคอมพิวเตอร์ แล้วแสดงผลออกที่จอคอมพิวเตอร์

2.3.การรับส่งข้อมูลผ่านคลื่นความถี่ในอากาศ

ใช้สายอากาศแบบ Wire whip antenna

- ย่านความถี่ 2.4 GHz
- รับส่งข้อมูลอัตราความเร็ว 250Kbps
- Indoor/Urban range: 133 ft(40 m)
- Outdoor/RF line-of-sight range:400 ft (120 m)
- Transmit power output:1.25 mW (+1 dBm) (2 mW(+3 dBm) boostmode)
- Receiver sensitivity (1% PER):-97 dBm (-98 dBm boost mode)

2.4.รูปแบบการเชื่อมต่ออุปกรณ์ (ไร้สาย)

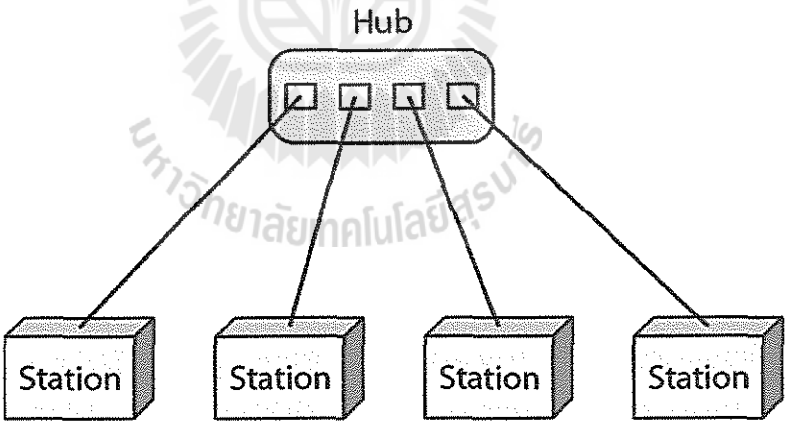
Point-to-Point

A Star topology connecting

เป็นการเชื่อมต่ออุปกรณ์หรือจุดต่าง ๆ ออกจากอุปกรณ์ศูนย์กลาง หรืออุปกรณ์แม่ข่ายที่เรียกว่า File Server แต่ละอุปกรณ์จะมีสัญญาณเชื่อมต่อกับอุปกรณ์ศูนย์กลาง ไม่มีการใช้สัญญาณร่วมกัน เมื่ออุปกรณ์ใดเกิดความเสียหาย จะไม่มีผลกระทบกับอุปกรณ์อื่น ๆ ปัจจุบันนิยมใช้อุปกรณ์ Hub เป็นตัวเชื่อมต่อ จากอุปกรณ์แม่ข่าย หรืออุปกรณ์ศูนย์กลาง

ข้อดี ของการเชื่อมต่อแบบนี้คือ ง่ายต่อการให้บริการ เพราะมีศูนย์กลาง อยู่ที่อุปกรณ์แม่ข่าย อยู่เครื่องเดียว และเมื่อเกิดความเสียหาย ที่อุปกรณ์เครื่องใดเครื่องหนึ่ง อุปกรณ์เครื่องอื่นก็จะไม่มีผลกระทบอันใด เพราะใช้สัญญาณคนละเส้นทาง

ข้อเสีย คือในกรณีที่เป็นสายต้องใช้สายสัญญาณจำนวนมาก เพราะแต่ละสถานี มีสายสัญญาณของตนเองเชื่อมต่อกับศูนย์กลาง จึงเหมาะกับเครือข่ายระยะใกล้ มากกว่าการเชื่อมต่อเครือข่ายระยะไกล การขยายระบบยุ่งยาก เพราะต้องเชื่อมสายจากศูนย์กลางออกมา ถ้าศูนย์กลางเสียหายระบบจะใช้การไม่ได้

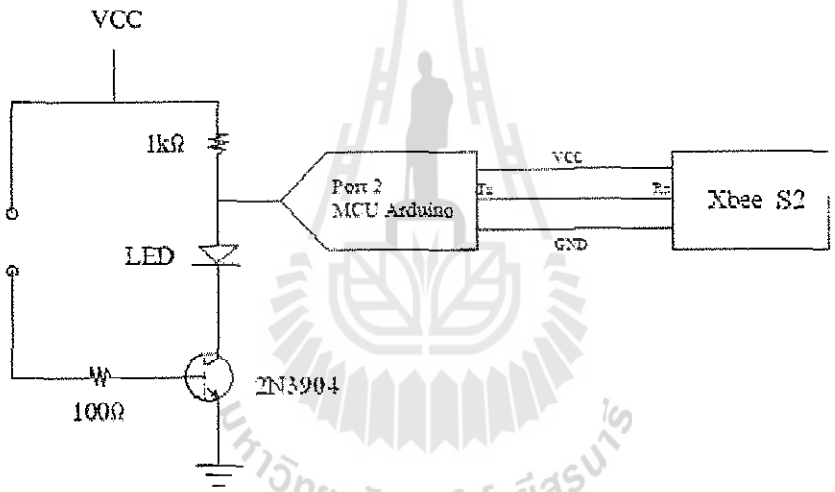


บทที่ 3
การออกแบบและการทำงานของระบบ

3.1 บทนำ

ในบทนี้จะกล่าวถึงขั้นตอนการออกแบบระบบการทำงานของเซ็นเซอร์ความชื้น การเชื่อมต่อเซ็นเซอร์ความชื้นกับอุปกรณ์สื่อสารไร้สาย(Xbee) และบอร์ดไมโครคอลโทรลเลอร์ (Arduino) การเขียนโปรแกรมควบคุมบอร์ดไมโครคอนโทรลเลอร์(Arduino)โดยใช้ภาษาซี การออกแบบโปรแกรมแสดงผล

3.2 เซ็นเซอร์ความชื้น



มหาวิทยาลัยเทคโนโลยีสุรนารี

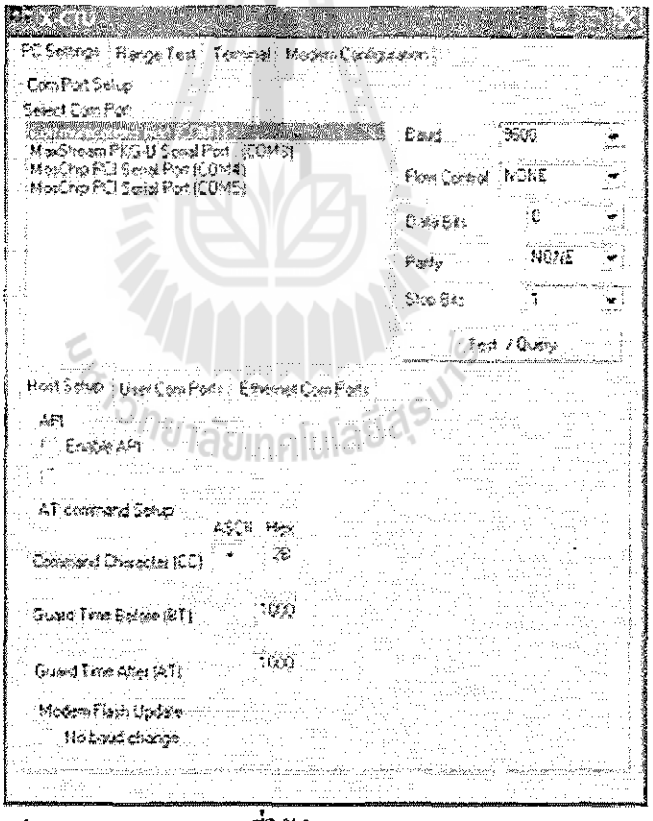
3.2 การเชื่อมต่อเซ็นเซอร์ความชื้นกับอุปกรณ์สื่อสารไร้สาย(Xbee)และคอมพิวเตอร์

การใช้งาน Xbeeเบื้องต้น

<u>รู้จัก X-CTU</u>

X-CTU เป็น software interface บนคอมพิวเตอร์ที่จะช่วยในการ update firmware หรือ ทดสอบการใช้งาน หรือ ปรับ parameter กับXbee โดยสามารถ download software มาได้ฟรีจาก Digi(X-CTU Software) สำหรับการใช้งานสามารถอ่านจากคู่มือ X-CTU Configuration & Test Utility Software User Guide

หลังจากที่ Download ตัว Software มาแล้ว การ Install จะใช้เวลานานพอสมควรเนื่องจาก จะมีการ Download Firmware ล่าสุดจาก Digi ผ่าน internet (ควรต่อ internet ไว้ด้วย) กรุณารอจน ติดตั้งเสร็จจะได้ firmware ครบถ้วน



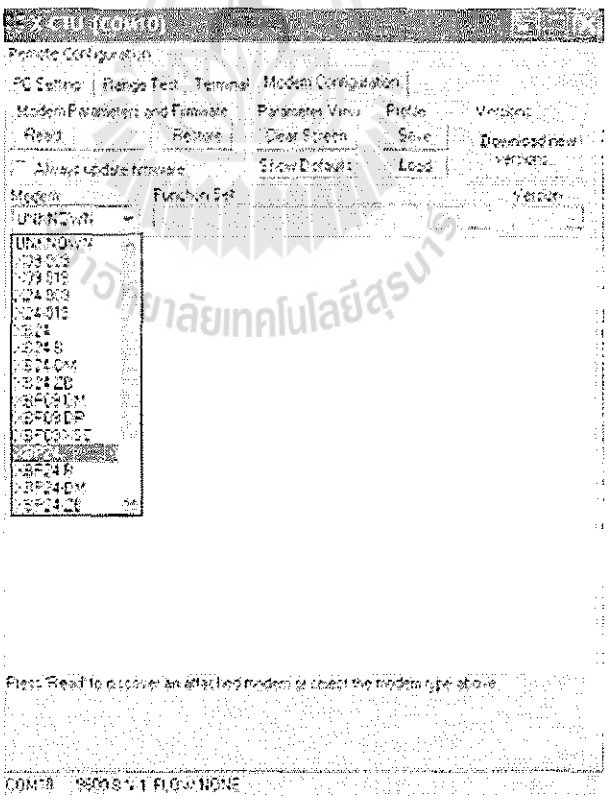
รูป Software X-CTU ที่ใช้ร่วมกับ Xbee (Free Download)

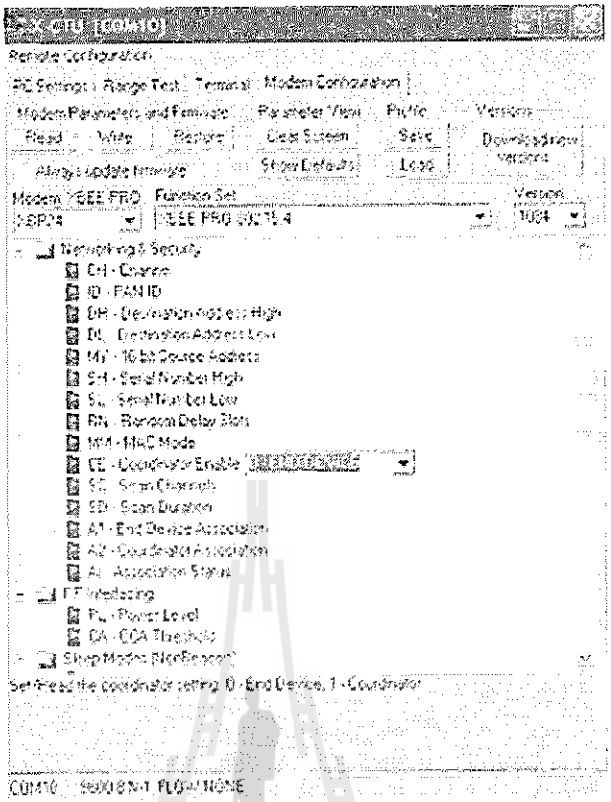
รู้จัก Firmware

สำหรับ Xbeeนั้น ในแต่ละรุ่นที่ซื้อ จะมี firmware ที่โปรแกรมมาแล้วจากทางโรงงาน ซึ่งเราสามารถดึงค่ามาดูได้โดยไปที่ Tab Modem Configuration แล้วกดปุ่ม Readหากซื้อ Xbee Pro Series 1 มา เราจะต้องเลือก firmware version ที่ใช้ร่วมกัน คือ 1081 , 1082 , 1083 ,1084 , ... ซึ่ง firmware ล่าสุดจะถูก set ให้โดยอัตโนมัติ (หาข้อมูลเรื่อง firmware ที่ใช้ในแต่ละรุ่น ได้จาก **www.digi.com** หรือจาก Datasheet)

หากเลือก Xbee รุ่นธรรมดา (ไม่ Pro) จะต้องเลือก group XB24xxxx เป็นต้นไป

ส่วนถ้าเลือกรุ่น Pro มา ต้องเลือก XBP24xxxxสำหรับ ในแต่ละ series นั้น จะลงท้ายไม่เหมือนกัน เช่น series2 ZB จะเป็น XB24-ZB





เราสามารถ Set parameter ใน Tab Modem Configuration ได้ แล้วกด write Firmware เราสามารถทดลองใช้งานด้วยการใช้อุปกรณ์ Dongle และ Xbee 1 คู่ สร้างเครือข่ายแบบ Point-to-Point ร่วมกับ X-CTU เพื่อกำหนด Parameter ให้กับ Xbeeผ่าน firmware และ X-CTU สามารถเปิดออกมาใช้กี่หน้าต่างก็ได้

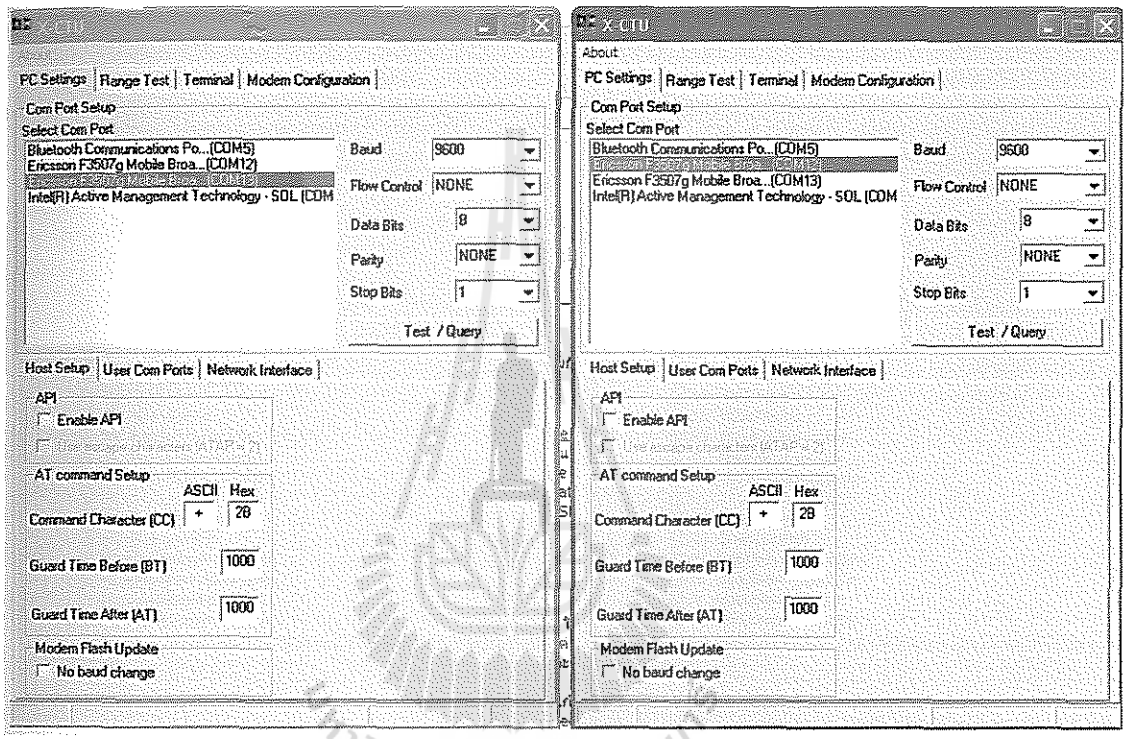
มหาวิทยาลัยเทคโนโลยีสุรนารี

<u>การ set อุปกรณ์</u>

การทดสอบจะใช้อุปกรณ์ดังรูปด้านล่างคือประกอบด้วย Xbee Pro Seires1 และ USB Dongle สำหรับเสียบอุปกรณ์กับคอมพิวเตอร์ (การทดสอบใช้เป็นคู่ ดังนั้นต้องใช้ Xbee Pro Seires1 และ USB Dongle ทั้งหมด 2 ชุด) ทำการเปิดX-CTU 2 หน้าต่าง แล้วไปที่หน้า Tab Modem Configuration แล้วกด Read จะพบว่า X-CTU จะทำการ Load Firmware ของ Xbee Pro Seires1 ออกมาซึ่งจะเป็นค่าที่ Set ไว้แบบ Default

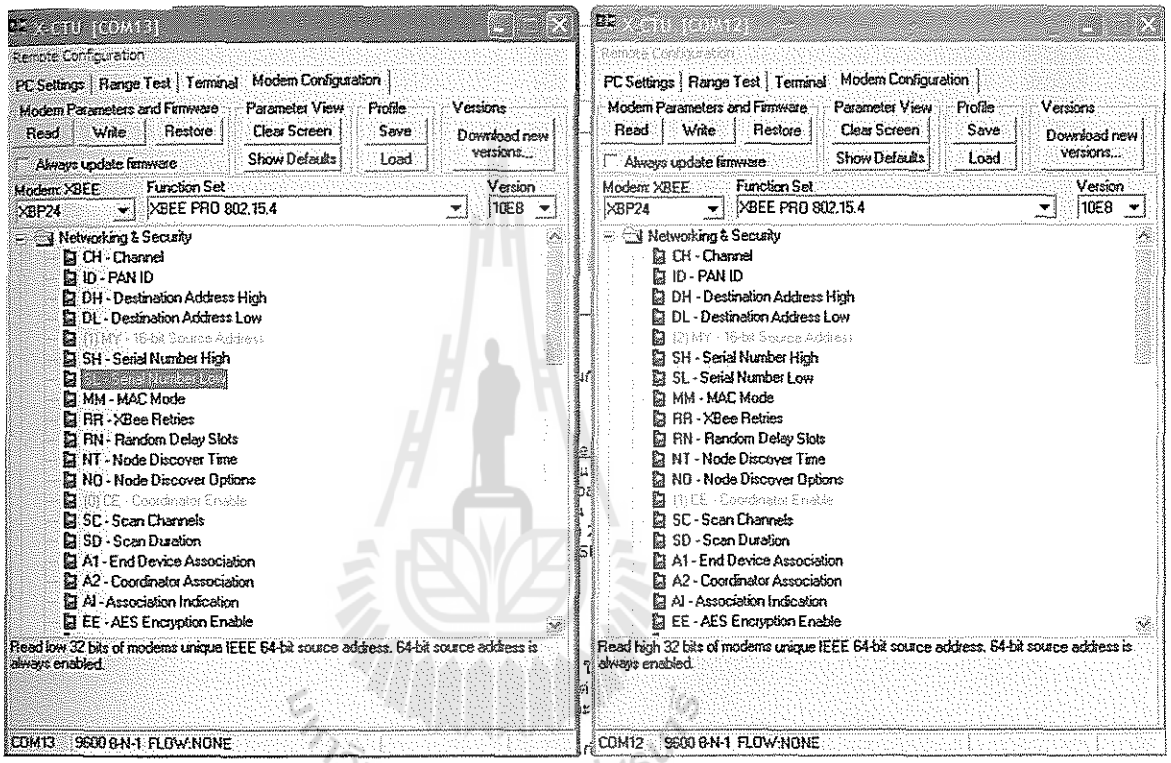
ทดสอบการใช้งาน Xbeeและ Software X-CTU

1. ทำการ set parameterให้ติดต่อกันแบบ Point to Point เมื่อทำการต่อ Xbee Pro Seires1 และ USB Dongle เข้ากับ PC ทั้ง 2 ชุดเรียบร้อยแล้วให้เปิดโปรแกรม X-CTU ขึ้น 2 ชุดเช่นกัน แล้วทำการเลือก COM Port (UART) ในแต่ละชุดให้ถูกต้อง



มหาวิทยาลัยเทคโนโลยีสุรนารี

2. ที่โปรแกรม X-CTU ให้เลือก Tab "Modem Configuration" แล้วทำการ set parameter โดยให้ฝั่งหนึ่งเป็น End Device ด้วยการ Set Parameter ในหมวด Networking & Security >> CE=0 และ ค่า MY = 1 และ ฝั่งอีกหนึ่ง เป็น Coordinator ด้วยการ Set Parameter ในหมวด Networking & Security >> (CE=1) และ ค่า MY = 2 ทั้งนี้ ได้ set baud rate ที่ตัว Xbeeทั้ง 2 ตัวที่ 9600 bps (BD=3)

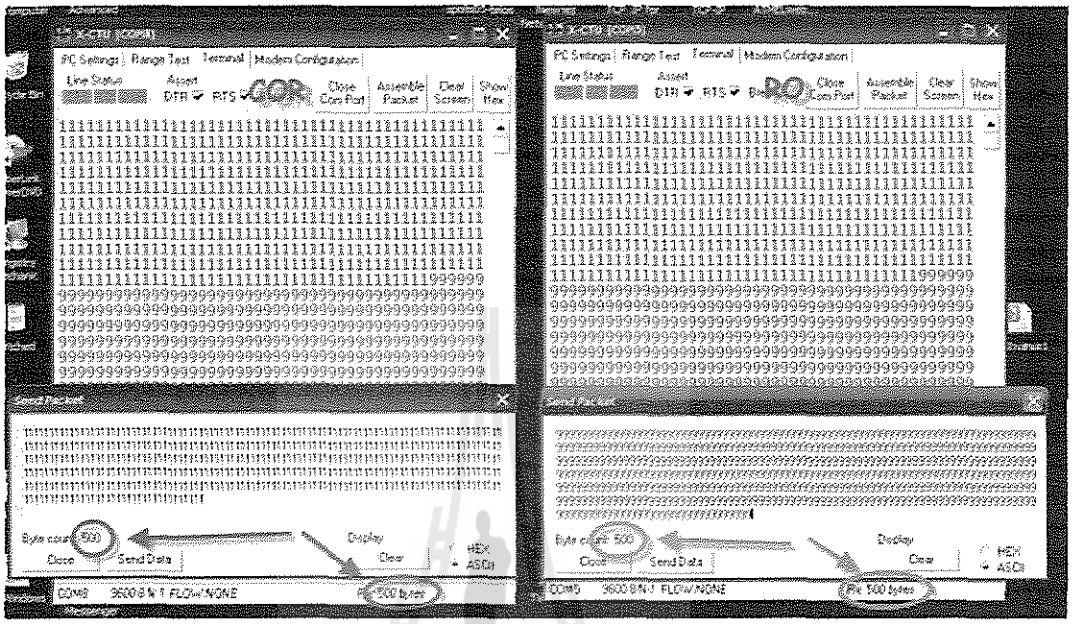


3. ทำการ set ให้ parameter DH และ DL ของแต่ละฝั่งให้มีค่าเท่ากับ SH และ SL ของฝั่งตรงข้าม โดย SH SL เป็นค่า address ที่เปลี่ยนไม่ได้ (Read Only) เป็นค่าที่ใส่มาจากโรงงาน

*** วิธีการดูว่า Xbeeของเรามีค่า SH และ SL เท่าไหร่ สามารถดูได้ 2 วิธีคือ ให้กด Read เพื่อ Load ค่าของ Xbeeเข้ามา จะทำให้เรา Click ดูค่า SH และ SL ได้ และวิธีที่ 2 คือดูจากทางด้านใต้ของตัว Xbeeจะมี Sticker แปะหมายเลข SH , SL มาให้แล้ว

4. กด write firmware แล้วทดสอบใช้งาน

นอกจากนี้ยังมี Tab Terminal ซึ่งถือว่าเป็น Tab ที่สามารถใช้ประโยชน์ในการ Debug กับสัญญาณข้อมูลที่รับส่งกันได้ เปรียบเสมือนโปรแกรม Serial Monitor



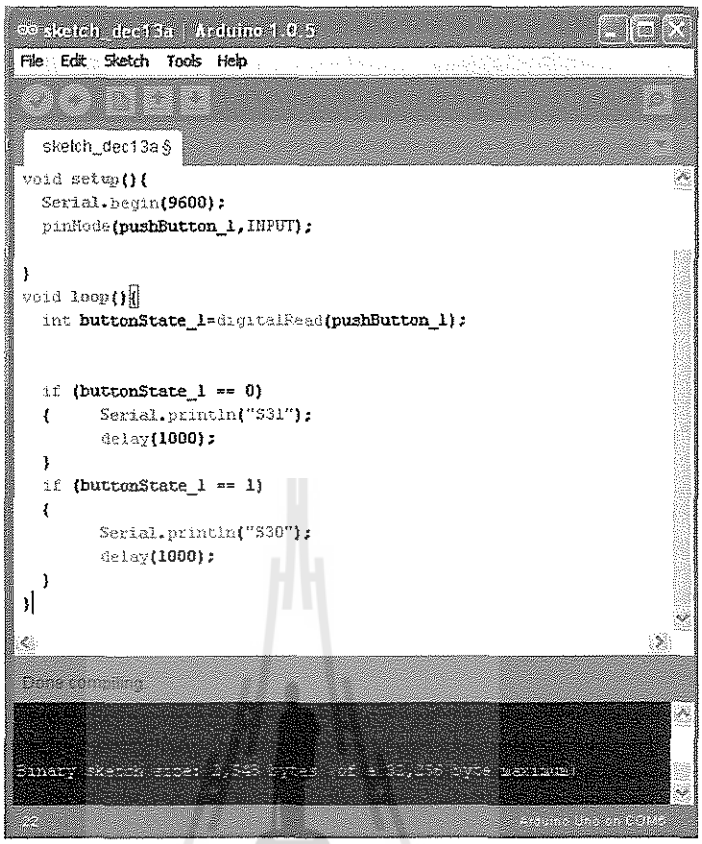
X_CTU นับข้อมูลแสดงเป็นตัวเลขให้ด้วย
(สีแดงคือข้อมูลที่รับได้ สีน้ำเงินคือข้อมูลที่ส่งออกไป)

มหาวิทยาลัยเทคโนโลยีสุรนารี

3.3การเขียนโปรแกรมควบคุมบอร์ดไมโครคอนโทรลเลอร์(Arduino)โดยใช้ภาษาซี

```
int pushButton_1=2;
int pushButton_2=3;
void setup(){
Serial.begin(9600);
pinMode(pushButton_1,INPUT);
}
void loop(){
int buttonState_1=digitalRead(pushButton_1);
if (buttonState_1 == 0)
{          Serial.println("S31");
delay(1000);
}
if (buttonState_1 == 1)
{
Serial.println("S30");
delay(1000);
}
}
```

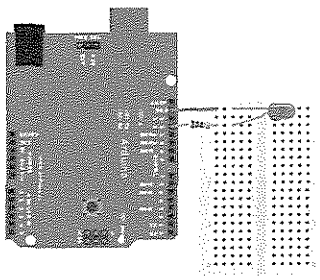
มหาวิทยาลัยเทคโนโลยีสุรนารี



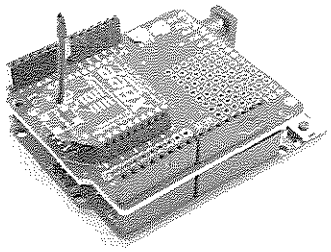
แนะนำ Microcontroller Arduino

Arduino อ่านว่า (อา-ดู-อิ-โน่ หรือ อาดุยโน่) เป็นบอร์ดไมโครคอนโทรเลอร์ตระกูล AVR ที่มีการพัฒนาแบบ Open Source คือมีการเปิดเผยข้อมูลทั้งด้าน Hardware และ Software ตัว บอร์ดถูกออกแบบมาให้ใช้งานได้ง่าย ดังนั้นจึงเหมาะสำหรับผู้เริ่มต้น ทั้งนี้ผู้ใช้งานยังสามารถดัดแปลงเพิ่มเติม พัฒนาต่อยอดทั้งตัวบอร์ด หรือโปรแกรมต่อได้อีกด้วย

ความง่ายของบอร์ด Arduino ในการต่ออุปกรณ์เสริมต่างๆ คือผู้ใช้งานสามารถต่อวงจรอิเล็กทรอนิคส์จากภายนอกแล้วเชื่อมต่อเข้ามาที่ขา I/O ของบอร์ด (ดูตัวอย่างรูปที่ 1) หรือเพื่อความสะดวกสามารถเลือกต่อกับบอร์ดเสริม (Shield) ประเภทต่างๆ (ดูตัวอย่างรูปที่ 2) เช่น XBee Shield, Music Shield, Relay Shield, Wireless Shield, GPRS Shield เป็นต้น มาเสียบกับบอร์ดบนบอร์ด Arduino แล้วเขียนโปรแกรมพัฒนาต่อได้เลย



รูปที่1 บอร์ด Arduino ต่อกับ LED



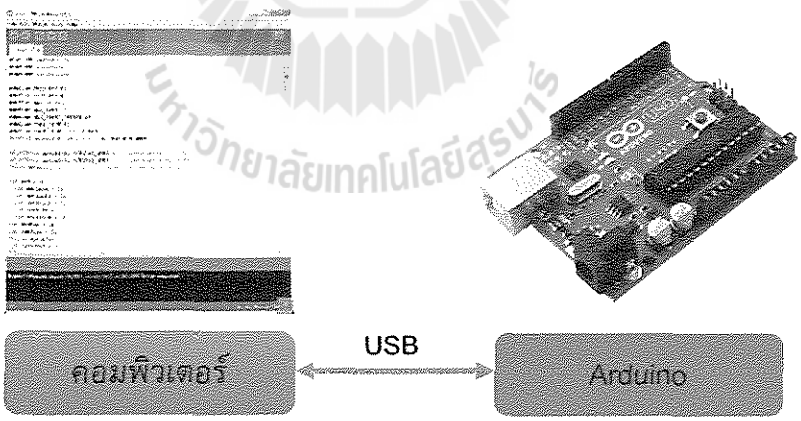
รูปที่2 บอร์ด Arduino ต่อกับบอร์ด XBee Shield

<u>จุดเด่นที่ทำให้บอร์ด Arduino เป็นที่นิยม</u>

ง่ายต่อการพัฒนา มีรูปแบบคำสั่งพื้นฐาน ไม่ซับซ้อนเหมาะสำหรับผู้เริ่มต้น มี Community กลุ่มคนที่ร่วมกันพัฒนาที่แข็งแรง Open Hardware ทำให้ผู้ใช้สามารถนำบอร์ดไปต่อยอดใช้งานได้หลายด้าน ราคาไม่แพง Cross Platform สามารถพัฒนาโปรแกรมบน OS ใดก็ได้

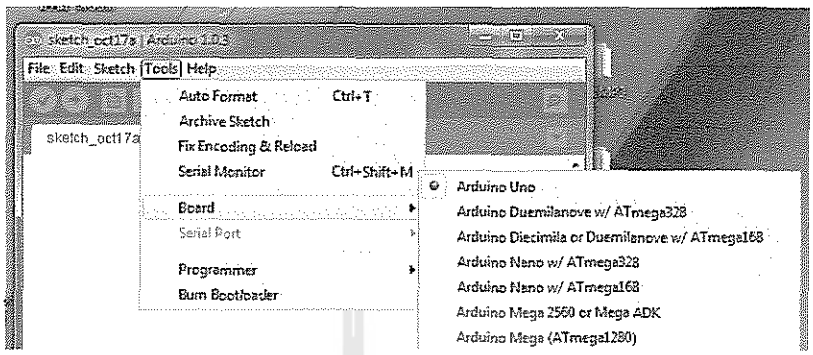
<u>รูปแบบการเขียนโปรแกรมบน Arduino</u>

1. เขียนโปรแกรมบนคอมพิวเตอร์ ผ่านทางโปรแกรม Arduino IDE ซึ่งสามารถดาวน์โหลดได้จาก <u>Arduino.cc/en/main/software</u>

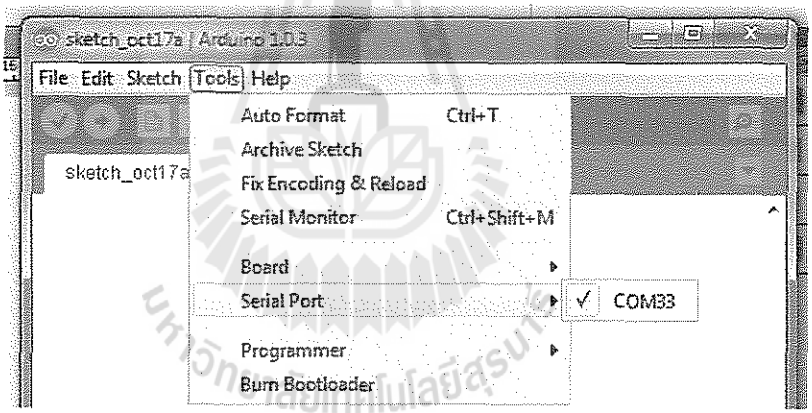


มหาวิทยาลัยเทคโนโลยีสุรนารี

2. หลังจากที่เขียนโค้ดโปรแกรมเรียบร้อยแล้ว ให้ผู้ใช้งานเลือกรุ่นบอร์ด Arduino ที่ใช้และหมายเลข Com port

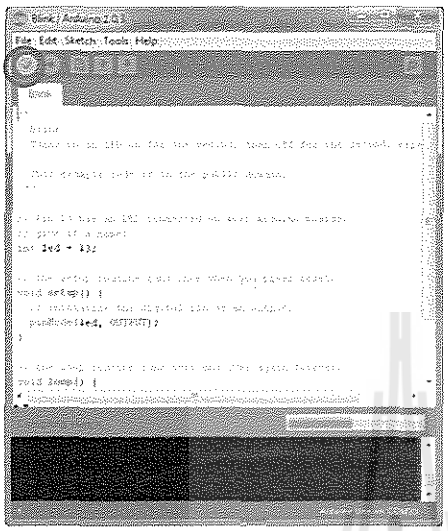


รูปที่ 3 เลือกรุ่นบอร์ด Arduino ที่ต้องการ upload

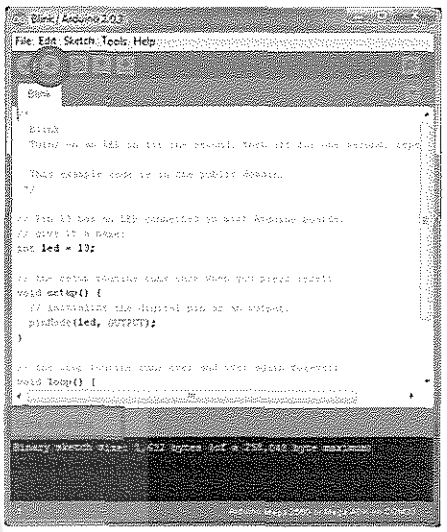


รูปที่ 4 เลือกหมายเลข Comport ของบอร์ด

3. กดปุ่ม Verify เพื่อตรวจสอบความถูกต้องและ Compile โค้ดโปรแกรม จากนั้นกดปุ่ม Upload โค้ด โปรแกรมไปยังบอร์ด Arduino ผ่านทางสาย USB เมื่ออับโหลดเรียบร้อยแล้ว จะแสดงข้อความแถบข้างล่าง “Done uploading” และบอร์ดจะเริ่มทำงานตามที่เขียนโปรแกรมไว้ได้ทันที



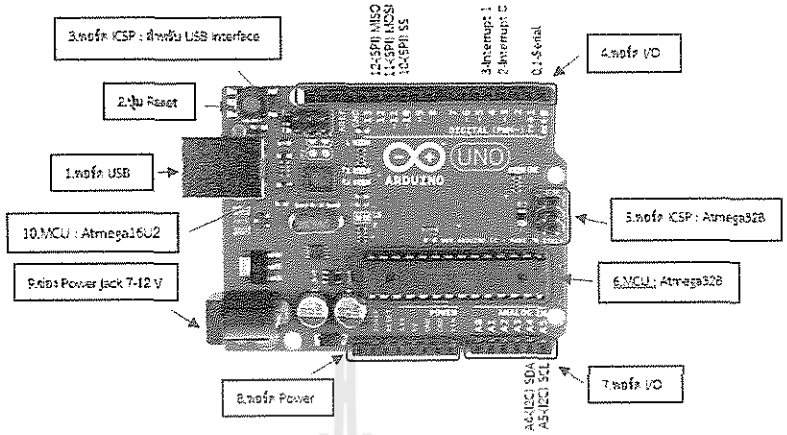
รูปที่5 กดปุ่ม Verify เพื่อตรวจสอบความถูกต้อง และ Compile โค้ดโปรแกรม



รูปที่6 Upload โค้ดโปรแกรม

มหาวิทยาลัยเทคโนโลยีสุรนารี

Layout & Pin out Arduino Board (Model: Arduino UNO R3)



1. USB Port: ใช้สำหรับต่อกับ Computer เพื่ออับโหลดโปรแกรมเข้า MCU และจ่ายไฟให้กับบอร์ด

2. Reset Button: เป็นปุ่ม Reset ใช้กดเมื่อต้องการให้ MCU เริ่มการทำงานใหม่

3. ICSP Port ของ Atmega16U2 เป็นพอร์ตที่ใช้โปรแกรม Visual Com port บน Atmega16U2

4. I/O Port: Digital I/O ตั้งแต่ขา D0 ถึง D13 นอกจากนี้ บาง Pin จะทำหน้าที่อื่นๆ เพิ่มเติมด้วย เช่น Pin 0,1 เป็นขา Tx,Rx Serial, Pin 3,5,6,9,10 และ 11 เป็นขา PWM

5. ICSP Port: Atmega328 เป็นพอร์ตที่ใช้โปรแกรม Bootloader

6. MCU: Atmega328 เป็น MCU ที่ใช้บนบอร์ด Arduino

7. I/O Port: นอกจากจะเป็น Digital I/O แล้ว ยังเปลี่ยนเป็น ช่องรับสัญญาณอนาล็อก ตั้งแต่ ขา A0-A5

8. Power Port: ไฟเลี้ยงของบอร์ดเมื่อต้องการจ่ายไฟให้กับวงจรภายนอก ประกอบด้วยขาไฟเลี้ยง +3.3 V, +5V, GND, Vin

9. Power Jack: รับไฟจาก Adapter โดยที่แรงดันอยู่ระหว่าง 7-12 V

10. MCU ของ Atmega16U2 เป็น MCU ที่ทำหน้าที่เป็น USB to Serial โดย Atmega328 จะติดต่อกับ Computer ผ่าน Atmega16U2

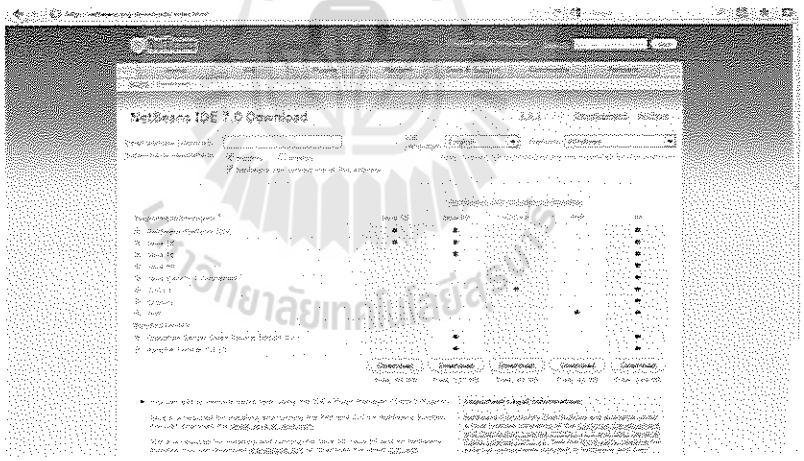
3.4 การออกแบบโปรแกรมแสดงผล

วิธีใช้ NetBeans เบื้องต้น

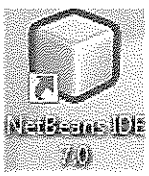
NetBeans IDE เป็น Freeware Editor อีกตัวหนึ่ง ซึ่งได้รับความนิยมในการใช้พัฒนาโปรแกรมโดย NetBeans นั้นสนับสนุนการพัฒนาโปรแกรมในหลายภาษา เช่นJava, php, C/C++ นอกจากนี้เรายังสามารถ download PlugIn ต่างๆมาใช้งานร่วมกับตัว NetBeans เองด้วย ยกตัวอย่างเช่น iReport เป็น Plugin ที่ใช้ในการสร้างreport หรือ Visual Web Page Layouts ใช้ในการแสดงตัวอย่างหน้า web page

ก่อนเริ่มใช้งาน NetBeans

ก่อนที่จะเริ่มใช้งาน NetBeans สามารถ download ตัวโปรแกรม NetBeans ได้ที่ http://netbeans.org/downloads/index.html โดย NetBeans นั้นมีหลายversion หลาย Platform สามารถเลือกภาษา Platform และ version ได้ตามความเหมาะสม



เมื่อทำการ download มาแล้วจะได้เป็น .exe file ให้ทำการติดตั้งโปรแกรมไปยัง directory ที่ต้องการจะทำการติดตั้งได้เลย หลังจากติดตั้งโปรแกรมเสร็จสมบูรณ์แล้ว เราสามามารถเข้าโปรแกรม NetBeans ได้จากหน้า desktop



ในที่นี้จะทำการพัฒนาโปรแกรมด้วยภาษา JAVA เพราะฉะนั้นจะต้องทำการ download ตัว JDK หรือ Java Development Kit เสียก่อน สามารถ downloadได้ที่

http://www.oracle.com/technetwork/java/javase/downloads/index.html

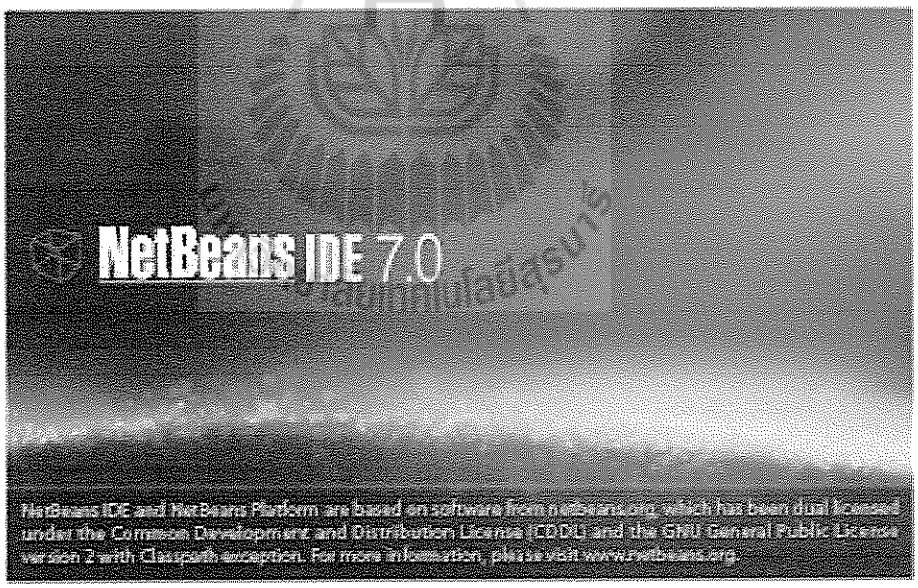
jdk-6u24-window
s-x64

เมื่อ download JDK แล้วก็ต้องทำการ Set class path เพื่อให้สามารถ Compile และ Run Java ได้ โดยวิธีทำสามารถติดตามได้จาก Link ด้านล่าง

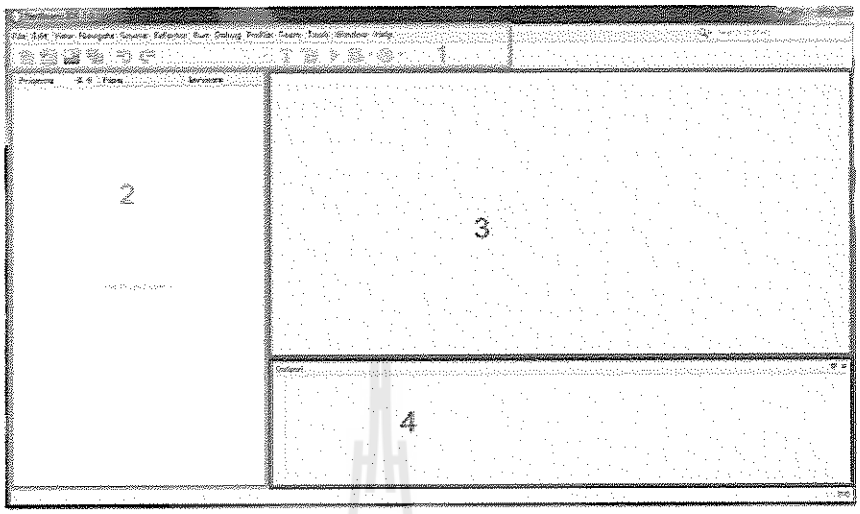
http://archive.oracle.in.th/2008/02/set-path-class-path-j2se-dos.html หรือ

http://www.dominixz.com/blog/software-tips/set-java-to-compile-and-run-all-places-in-dos/

เริ่มใช้งาน NetBeans



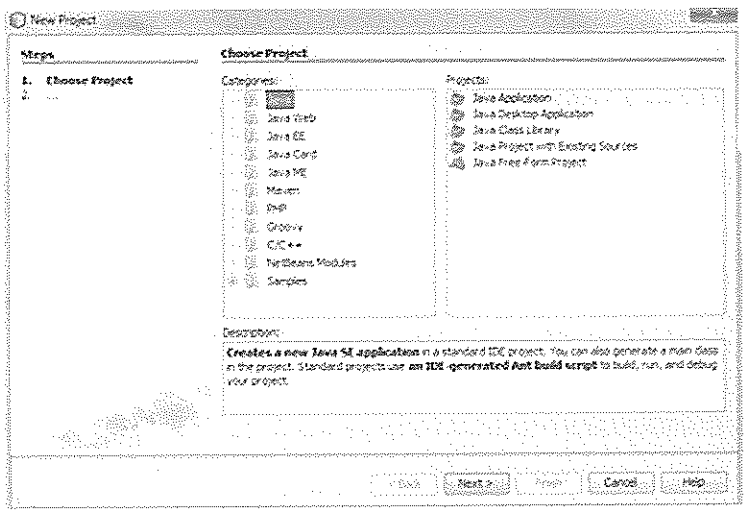
หน้าจอหลักของ NetBeans จะแบ่งออกเป็น 4 ส่วนใหญ่ๆดังนี้



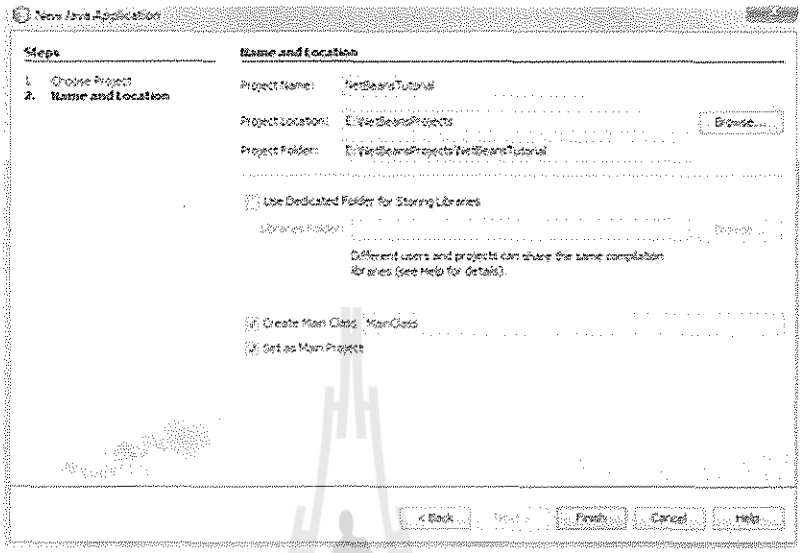
1. เป็นส่วนของแถบเมนู และ Tool ต่างๆ
2. เป็นส่วนที่แสดงและจัดการ Project ต่างๆ เหมือนเป็นการ Browse ดู Projectหรือ File ต่างๆ
3. เป็นส่วนที่ใช้ในการเขียน Code
4. เป็นส่วนที่ใช้ในการแสดงผลการทำงานต่าง เมื่อทำการ Run โปรแกรม และยังมีส่วนของการ debug โปรแกรมด้วย

การเขียนโปรแกรม

โดยเริ่มจากการ New Project ใหม่ขึ้นมา โดยการเลือกที่ File >> New Project

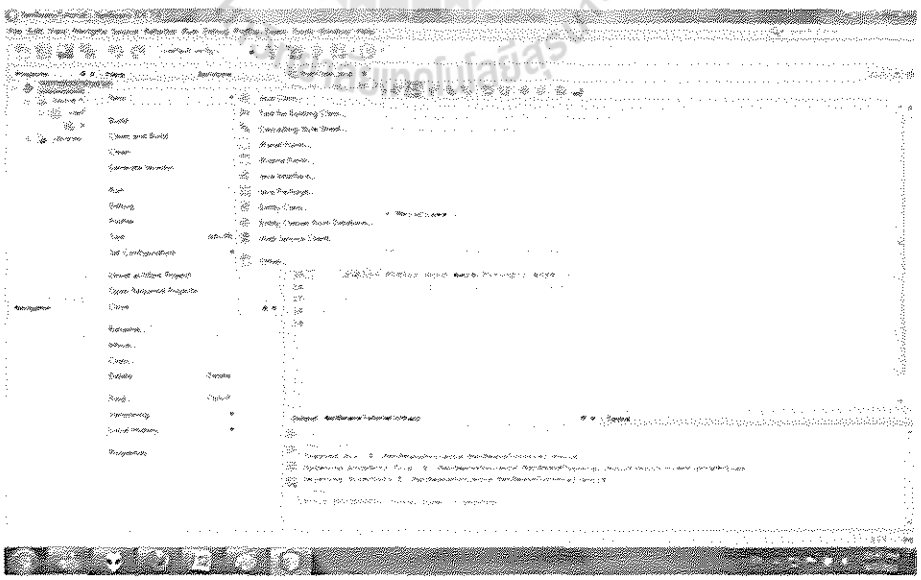


เลือก Java และ Java Application >> Next

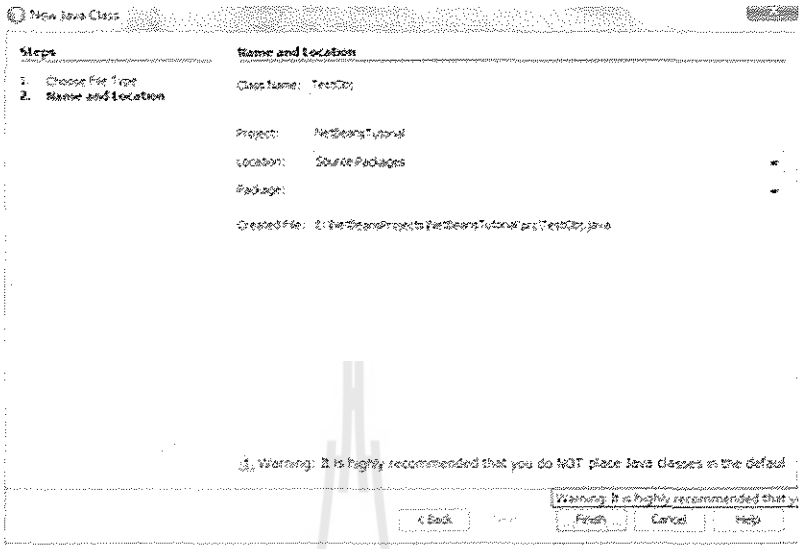


จากนั้นจะมีหน้าต่างให้กำหนดค่ารายละเอียดต่างๆของ Project โดยจะบังคับให้ใส่ Project Name >> Finish

สร้าง class TestObj เพื่อใช้ Funtion บางอย่างของ NetBeans คลิกขวาที่ชื่อ Project แล้วเลือกNew >> Java Class ดังภาพ



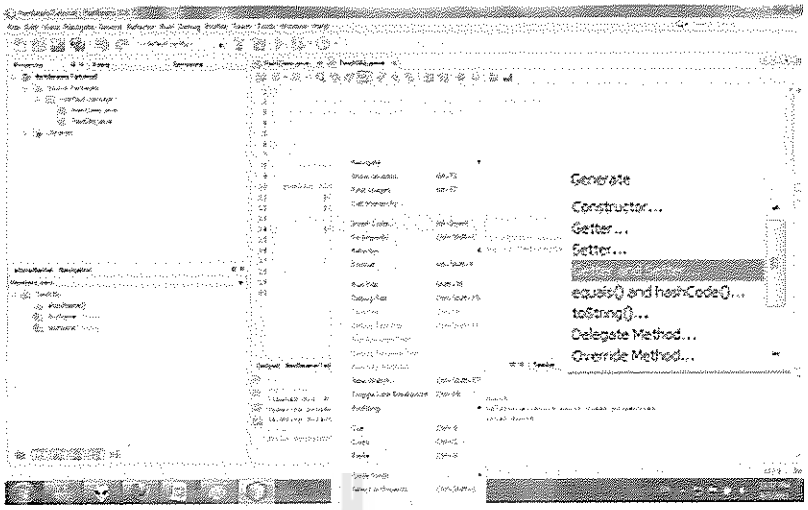
จะปรากฏหน้าต่างขึ้นมาให้กำหนดรายละเอียด โดยให้ตั้งชื่อ Class ว่า TestObj >> Finish



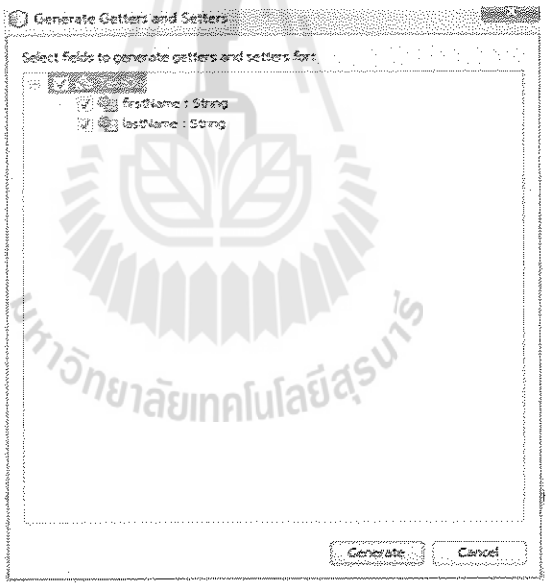
class TestObj จะมี code ดังนี้ครับ

```
public class TestObj {
private String firstName;
private String lastName;
public void showName() {
System.out.println("THIS IS METHOD showName()");
System.out.println("My name is "+firstName+" "+lastName);
        }
}
```

จะเห็นได้ว่า Attribute ใน Class TestObj เป็น private ฉะนั้นก็จะต้องมี Getter and Setter สามารถที่จะสั่งให้ NetBeans สร้าง Getter and Setter จากAttribute ที่มีอยู่ได้โดยการคลิกขวาที่หน้าจอที่ใช้เขียน code แล้วเลือก Insert Codeจะปรากฏหน้าต่างขึ้นมาให้เลือกที่ Getter and Setter



หน้าต่างสำหรับการ Generate Getters and Setters จะขึ้นมา ให้ทำการเลือกว่าจะสร้าง Getters and Setters ให้กับ Attribute ใดบ้าง เมื่อเลือกเสร็จเรียบร้อยให้กด Generate



เราจะได้ code ของ class TestObj ดังนี้

```
public class TestObj {
private String firstName;
private String lastName;
public void showName() {
System.out.println("THIS IS METHOD showName()");
```

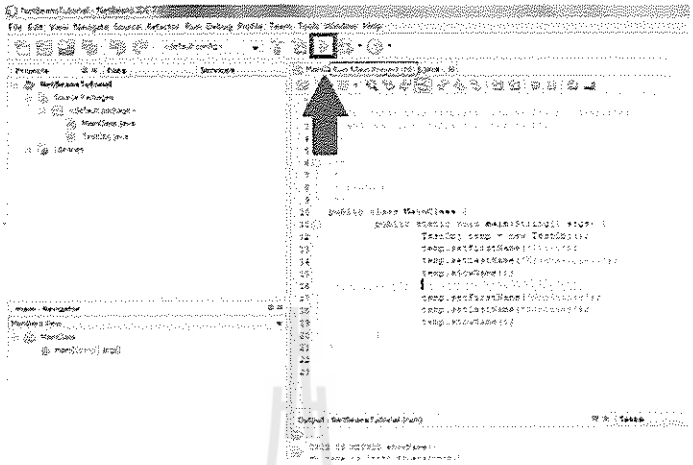
```
System.out.println("My name is "+firstName+" "+lastName);
}
public String getFirstName() {
        return firstName;
}
public void setFirstName(String firstName) {
        this.firstName = firstName;
}
public String getLastName() {
return lastName;
}
public void setLastName(String lastName) {
this.lastName = lastName;
}
}
```

นอกจากการ Generate Getters and Setters แล้ว สามารถคลิกขวาเพื่อดูFunction อื่นๆที่ NetBeans สามารถช่วยจัดการให้การทำงานง่ายมากขึ้น เมื่อได้ class TestObj แล้ว ต่อไปทำการแก้ไข code ของ MainClass ให้เป็นดังนี้

```
public class MainClass

public static void main(String[] args) {
TestObj temp = new TestObj();
temp.setFirstName("Itthi");
temp.setFirstName("Itthi");
temp.showName();
temp.setFirstName("Monthinee");
temp.setLastName("Buntawee");
temp.showName();
}
}
```

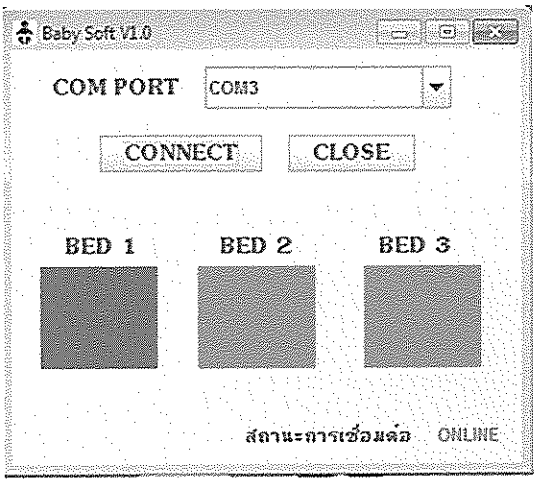
มหาวิทยาลัยเทคโนโลยีสุรนารี

เมื่อทำการแก้ไข Code เสร็จเรียบร้อย ทำการ Run โปรแกรมโดยการกดปุ่มสีเขียวที่อยู่ด้านล่างแถบเมนู



โดย NetBeans จะทำการ Save ให้อัตโนมัติทุกครั้งที่จะทำการ Run ถ้า NetBean sทำการ compile แล้วไม่พบ error โปรแกรมจะแสดงผลที่ส่วนด้านล่างดังภาพ

จากนั้นให้ทำการเปิดโปรแกรมใหม่ ใช้ Code ในภาคผนวก ข. ส่วนที่1 และสอง ในการรันโปรแกรม จะได้โปรแกรมการแสดงผลดังรูป



บทที่ 4

ผลการทดลอง

4.1 บทนำ

ในบทนี้จะกล่าวถึงขั้นตอนการทดลองและผลการทดลอง

4.2 ผลการทดลอง

การทดลองปริมาณของของเหลวที่ระยะห่างของสายไฟ 10 cm. (แผ่นรองเปื้อน)

เทน้ำครั้งที่	ปริมาณน้ำ (ml.)	การแสดงผลที่จอมอนิเตอร์
1	20	ไม่แสดงผล
2	40	ไม่แสดงผล
3	60	แสดงผล
4	80	แสดงผล
5	100	แสดงผล

มหาวิทยาลัยเทคโนโลยีสุรนารี

การทดลองปริมาณของของเหลวที่ระยะห่างของสายไฟ 15 cm. (แผ่นรองเปื้อน)

เทน้ำครั้งที่	ปริมาณน้ำ (ml.)	การแสดงผลที่จอมอนิเตอร์
1	20	ไม่แสดงผล
2	40	ไม่แสดงผล
3	60	ไม่แสดงผล
4	80	แสดงผล
5	100	แสดงผล

การทดลองปริมาณของของเหลวที่ระยะห่างของสายไฟ 5 cm. (ผ้าอ้อมผู้ใหญ่)

เทน้ำครั้งที่	ปริมาณน้ำ (ml.)	การแสดงผลที่จอมอนิเตอร์
1	20	แสดงผล
2	40	แสดงผล
3	60	แสดงผล
4	80	แสดงผล
5	100	แสดงผล

มหาวิทยาลัยเทคโนโลยีสุรนารี

การทดลองปริมาณของของเหลวที่ระยะห่างของสายไฟ 10 cm. (ผ้าอ้อมผู้ใหญ่)

เทน้ำครั้งที่	ปริมาณน้ำ (ml.)	การแสดงผลที่จอมอนิเตอร์
1	20	ไม่แสดงผล
2	40	ไม่แสดงผล
3	60	แสดงผล
4	80	แสดงผล
5	100	แสดงผล

การทดลองปริมาณของของเหลวที่ระยะห่างของสายไฟ 3 cm. (ผ้าอ้อมเด็ก)

เทน้ำครั้งที่	ปริมาณน้ำ (ml.)	การแสดงผลที่จอมอนิเตอร์
1	10	แสดงผล
2	15	แสดงผล
3	20	แสดงผล
4	25	แสดงผล
5	30	แสดงผล

มหาวิทยาลัยเทคโนโลยีสุรนารี

การทดลองปริมาณของของเหลวที่ระยะห่างของสายไฟ 5 cm. (ผ้าอ้อมเด็ก)

เทน้ำครั้งที่	ปริมาณน้ำ (ml.)	การแสดงผลที่จอมอนิเตอร์
1	10	ไม่แสดงผล
2	15	ไม่แสดงผล
3	20	แสดงผล
4	25	แสดงผล
5	30	แสดงผล

4.3.วิเคราะห์ผลการทดลอง

จากการทดลองทำให้เห็นว่าระยะห่างระหว่างสายไฟมีผลต่อปริมาณที่ทำให้อุปกรณ์ทำงาน ในเมื่อทดสอบระยะห่างระหว่างสายไฟ โดยใช้ปริมาณน้ำที่เท่ากัน ถ้าระยะห่างระหว่างสายไฟมากเกินไปอุปกรณ์ทำงานช้ากว่าความต้องการเปลี่ยนแผ่นลองหรือผ้าอ้อม ในทางกลับกันระยะห่างน้อยเกินไปอุปกรณ์ทำงานเร็วกว่าความต้องการเปลี่ยนแผ่นลองหรือผ้าอ้อม

จากการทดลองระยะห่างระหว่างสายไฟในแผ่นลองเปื้อนอยู่ที่ 10 cm. อุปกรณ์ทำงานที่ปริมาณน้ำ60ml ในผ้าอ้อมผู้ใหญ่อยู่ที่10 cm. อุปกรณ์ทำงานที่ปริมาณน้ำ60ml. ในผ้าอ้อมเด็กอยู่ที่ 5cm. อุปกรณ์ทำงานที่ปริมาณน้ำ20ml. . ซึ่งเป็นปริมาณโดยเฉลี่ยที่เหมาะสมกับการเปลี่ยนแผ่นลองใหม่และผ้าอ้อมใหม่ บางขณะขึ้นอยู่กับการซึมซับของผ้าอ้อมด้วย

บทที่ 5
สรุป อภิปรายผล และข้อเสนอแนะ

5.1 สรุปผลการทดลอง

จากผลการทดลองพบว่าการที่จะทำให้ระบบตรวจจับความชื้นในผ้าอ้อมมีความแม่นยำในการทำงานนั้นขึ้นอยู่กับปัจจัยดังต่อไปนี้

1. การกำหนดระยะห่างระหว่างสายไฟ อย่างเหมาะสมกับผ้าอ้อม และบุคคลที่ใช้งาน

2. ความเหมาะสมในการเลือกใช้งาน

จะเห็นได้ว่า เมื่อกำหนดระยะห่างระหว่างสายไฟได้เหมาะสมกับปริมาณที่จะทำให้ต้องการเปลี่ยนผ้าอ้อมนั้น จะทำให้อุปกรณ์สามารถแสดงผลได้อย่างตรงตามความต้องการของผู้ใช้

5.2. สิ่งที่ได้รับจากการทำโครงงาน

1.ขั้นตอนวิธีการใช้งาน Xbee และบอร์ด Arduio

2.ศึกษาการเขียนโปรแกรมด้วยภาษาซี

3.สามารถนำระบบตรวจวัดความชื้นไปใช้ได้จริง

4.เรียนรู้การทำงานเป็นทีม

5.3.ปัญหาและอุปสรรค

1.ขนาดของบอร์ดใหญ่เกินไป ทำให้อุปกรณ์โดยรวมมีขนาดใหญ่ขึ้นด้วย

2.ผู้ร่วมโครงงานไม่มีความถนัดด้านการเขียนโปรแกรมโดยใช้ภาษซี และ java จำเป็นต้องศึกษาเพิ่มเติมทำให้กระทบกับเวลาในการจัดทำส่วนอื่น

3.โปรแกรมแสดงผลกำหนดจำนวนผู้ใช้งาน หากต้องการเพิ่มจำนวนต้องทำการแก้ไขส่วน Code Program จึงทำให้ไม่สะดวกในการใช้งานมากนัก

5.4.ข้อเสนอแนะ

1.การแสดงผลขึ้นอยู่กับการซึมซับของผ้าอ้อมแต่ละยี่ห้อ และอิริยาบถต่างๆ

2.ควรใช้แบตเตอรี่แทนถ่าน เพื่อการใช้งานที่ยาวนานขึ้น

3.การแสดงผลอาจมีการเพิ่มเสียงเตือน สามารถรับทราบได้มาไม่ได้อยู่หน้าจอแสดงผล

เอกสารอ้างอิง

1. http://www.arduino.cc/en/Main/Policy
2. http://portal.in.th/eleccom69/pages/6181/
3. http://www.thaieasyelec.com/Review-Product-Article/step-by-step-to-use-xbee-from-digi.html
4. http://www.thaieasyelec.com/Embedded-Electronics-Application/Xbee-Basic-Configuration-in-Network-Application.html
5. http://www.ec.in.th/index.php?route=cms%2Farticle&article_id=32

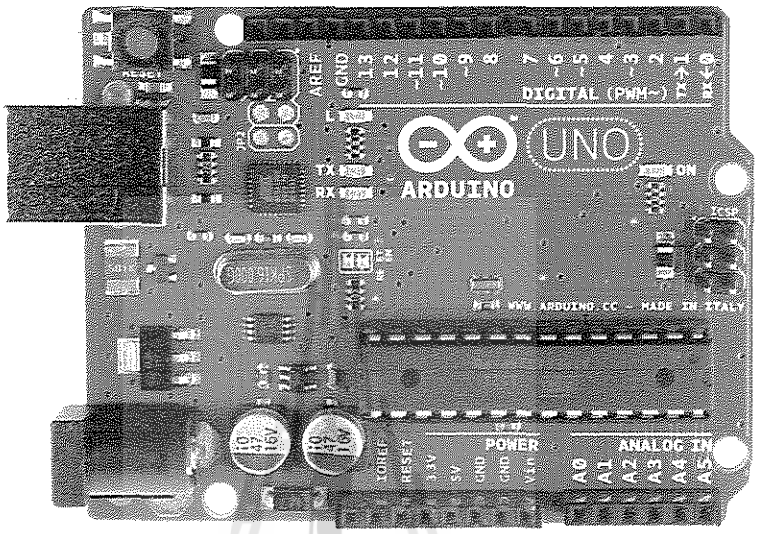
มหาวิทยาลัยเทคโนโลยีสุรนารี

ภาคผนวก

มหาวิทยาลัยเทคโนโลยีสุรนารี

ภาคผนวก ก

บอร์ดไมโครคอนโทรลเลอร์(Arduino)



Core Functions

Simple programs that demonstrate basic Arduino commands. These are included with the Arduino environment; to open them, click the Open button on the toolbar and look in the examples folder.

Basics

BareMinimum: The bare minimum of code needed to start an Arduino sketch.

Blink: Turn an LED on and off.

DigitalReadSerial: Read a switch, print the state out to the Arduino Serial Monitor.

AnalogReadSerial: Read a potentiometer, print it's state out to the Arduino Serial Monitor.

Fade: Demonstrates the use of analog output to fade an LED.

ReadAnalogVoltage : Reads an analog input and prints the voltage to the serial monitor

มหาวิทยาลัยเทคโนโลยีสุรนารี

Digital

Blink Without Delay: blinking an LED without using the delay() function.

Button: use a pushbutton to control an LED.

Debounce: read a pushbutton, filtering noise.

Button State Change: counting the number of button pushes.

Input Pullup Serial: Demonstrates the use of INPUT_PULLUP with pinMode().

Tone: play a melody with a Piezo speaker.

Pitch follower: play a pitch on a piezo speaker depending on an analog input.

Simple Keyboard: a three-key musical keyboard using force sensors and a piezo speaker.

Tone4: play tones on multiple speakers sequentially using the tone() command.

Analog

AnalogInOutSerial: read an analog input pin, map the result, and then use that data to dim or brighten an LED.

Analog Input: use a potentiometer to control the blinking of an LED.

AnalogWriteMega: fade 12 LEDs on and off, one by one, using an Arduino Mega board.

Calibration: define a maximum and minimum for expected analog sensor values.

Fading: use an analog output (PWM pin) to fade an LED.

Smoothing: smooth multiple readings of an analog input.

Communication

These examples include code that allows the Arduino to talk to Processing sketches running on the computer. For more information or to download Processing, see processing.org. There are also Max/MSP patches that can communicate with each Arduino sketch as well. For more on Max/MSP see Cycling 74. For Pd patches that can communicate with these sketches, see Scott Fitzgerald's examples.

ReadASCIIString: parse a comma-separated string of ints to fade an LED

ASCII Table: demonstrates Arduino's advanced serial output functions.

Dimmer: move the mouse to change the brightness of an LED.

Graph: send data to the computer and graph it in Processing.

Physical Pixel: turn a LED on and off by sending data to your Arduino from Processing or Max/MSP.

Virtual Color Mixer: send multiple variables from Arduino to your computer and read them in Processing or Max/MSP.

Serial Call Response: send multiple vairables using a call-and-response (handshaking) method.

Serial Call Response ASCII: send multiple variables using a call-and-response (handshaking) method, and ASCII-encode the values before sending.

SerialEvent: Demonstrates the use of SerialEvent(). Serial input (Switch (case) Statement): how to take different actions based on characters received by the serial port.

MIDI: send MIDI note messages serially.

MultiSerialMega: use two of the serial ports available on the Arduino Mega.

Control Structures

If Statement (Conditional): how to use an if statement to change output conditions based on changing input conditions.

For Loop: controlling multiple LEDs with a for loop and.

Array: a variation on the For Loop example that demonstrates how to use an array.

While Loop: how to use a while loop to calibrate a sensor while a button is being read.

Switch Case: how to choose between a discrete number of values. Equivalent to multiple If statements. This example shows how to divide a sensor's range into a set of four bands and to take four different actions depending on which band the result is in.

Switch Case 2: a second switch-case example, showing how to take different actions based in characters received in the serial port.

Sensors

ADXL3xx: read an ADXL3xx accelerometer.

Knock: detect knocks with a piezo element.

Memsic2125 : two-axis acceleromoter.

Ping: detecting objects with an ultrasonic range finder.

Display

Examples of basic display control

LED Bar Graph: how to make an LED bar graph.

Row Column Scanning: how to control an 8x8 matrix of LEDs.

Strings

StringAdditionOperator: add strings together in a variety of ways.

StringAppendOperator: append data to strings.

StringCaseChanges: change the case of a string.

StringCharacters: get/set the value of a specific character in a string.

StringComparisonOperators: compare strings alphabetically.

StringConstructors: how to initialize string objects.

StringIndexOf: look for the first/last instance of a character in a string.

StringLength&StringLengthTrim: get and trim the length of a string.

StringReplace: replace individual characters in a string.

StringStartsWithEndsWith: check which characters/substrings a given string starts or ends with.

StringSubstring: look for "phrases" within a given string.

USB (Leonardo, Micro, and Due specific examples)

The Keyboard and Mouse examples are unique to the Leonardo, Micro and Due. They demonstrate the use of libraries that are unique to the board.

KeyboardAndMouseControl: Demonstrates the Mouse and Keyboard commands in one program.

Keyboard

KeyboardMessage: Sends a text string when a button is pressed.

KeyboardLogout : Logs out the current user with key commands

KeyboardSerial: Reads a byte from the serial port, and sends back a keystroke.

KeyboardReprogram : opens a new window in the Arduino IDE and reprograms the Leonardo with a simple blink program

Mouse

ButtonMouseControl: Control cursor movement with 5 pushbuttons.

JoystickMouseControl: Controls a computer's cursor movement with a Joystick when a button is pressed.

มหาวิทยาลัยเทคโนโลยีสุรนารี

Use of the ARDUINO trademark must be compliant with http://www.arduino.cc/en/Main/Policy

ARDUINO is a registered trademark.

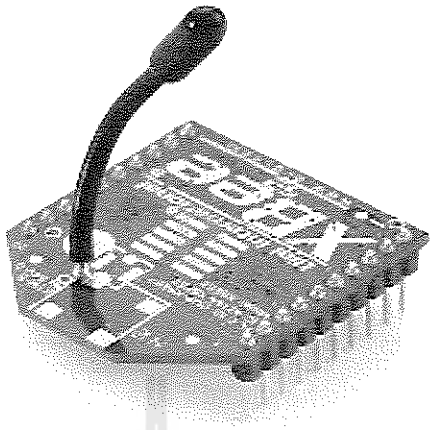
The product information on the Web site or Materials is subject to change without notice. Do not finalize a design with this information.

Use for future design and shall have no responsibility whatsoever for customers' products and applications with respect to any such use.

ARDUINO PRODUCTS ARE NOT AUTHORIZED FOR USE IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT EXPRESS WRITTEN APPROVAL OF ARDUINO.

Design Reference ARE PROVIDED "AS IS" AND "WITH ALL FAULTS". ANY EXPRESS OR IMPLIED WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE AND MERCHANTABILITY AND FITNESS FOR ANY PARTICULAR PURPOSE (INCLUDING, WITHOUT LIMITATION, IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE) ARE DISCLAIMED. IN NO EVENT SHALL ARDUINO BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, LOST PROFITS) ARISING IN ANY WAY OUT OF THE DESIGN REFERENCE OR ANY ARDUINO PRODUCTS, INCLUDING BUT NOT LIMITED TO, ANY DEFECT THEREIN, INFRINGEMENT, LOST PROFITS OR ANY SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES RESULTING FROM DESIGN REFERENCE OR ARDUINO PRODUCTS USE, HOWEVER CAUSED AND WHETHER ARISING UNDER CONTRACT, TORT OR OTHER LEGAL THEORY, EVEN IF ARDUINO HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Arduino may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Arduino reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

Xbee BASIC



Xbee เป็นอุปกรณ์ที่มี Microcontroller และ RF IC อยู่ภายใน (อ้างอิงได้จาก Schematic) ทำหน้าที่เป็น อุปกรณ์ transceiver (อุปกรณ์รับ-ส่งสัญญาณ) แบบ แบบ Half Duplex ย่านความถี่ 2.4 Ghz มีการจัดการโดยใช้พลังงานต่ำ ใช้งานง่าย มี interface ที่ใช้รับและส่งข้อมูล กับ Xbee เป็น UART (TTL) ซึ่งสำหรับทางด้านไมโครคอนโทรลเลอร์ เรานำขาที่ใช้ติดต่อสื่อสาร UART ของXbee ต่อเข้ากับ UART ของไมโครคอนโทรลเลอร์ได้เลยXbee สามารถใช้งานตามมาตรฐาน Zigbee ได้ โดยที่ท่านไม่ต้องเขียนโปรแกรมสร้าง เครือข่าย Zigbee เลย เพราะว่าทางผู้ผลิตได้จัดทำ firmware ที่จะโหลดเข้าไปในตัว Xbee ให้เราสามารถ set parameter ผ่าน software interface (X-CTU หรือโปรแกรมที่เขียนขึ้นเอง) , ผ่านทาง At command (เหมือนกับการควบคุม GSM Module) โดยใช้ Hyper terminal หรือ ผ่านทางการรับส่งข้อมูลด้วยไมโครคอนโทรลเลอร์ ได้อย่างง่ายดาย โดยเมื่อ set Xbee ให้ทำงานเป็นอุปกรณ์ในเครือข่าย Zigbee แล้ว เราจะเรียกXbee แต่ละตัวว่าเป็น Node (parameter ของ Xbee นั้นมีหลายตัว)

Firmware ที่ใช้กับ Xbeeจะใช้โหลดผ่านโปรแกรม X-CTU ซึ่งจะกล่าวถึงวิธีการใช้งานในบทความถัดไป (การใช้งาน Xbeeเบื้องต้น) ทั้งนี้ Xbeeแต่ละรุ่น จะสามารถ Setting Function การใช้งานได้มากมาย ทำให้ Firmware ที่จะต้องโหลดเข้าไปนั้น มีมากมายหลายแบบ เราต้องเลือกให้เหมาะสมกับการใช้งานเอง

Feature Summary ของ Xbee โดยรวมที่เหมือนกัน

1. Operating Frequency ISM Band 2.4 Ghz

(ISM Band หมายถึง ย่านความถี่ใช้งานเพื่อการวิจัย ซึ่งจะอนุญาตให้ใช้กับ อุตสาหกรรม (Industrial) วิทยาศาสตร์ (Scientific)และ ทางการแพทย์ (Medical) รวมเป็น ISM)

2. มีสายอากาศให้เลือกใช้หลายแบบ คือ แบบ Chip Ant , Whip Ant , UFL con , RPSMA con โดย 2 แบบหลัง เราต้องไปหาเสาอากาศย่าน 2.4 Ghz ที่เป็น connector แบบ UFL หรือ SMA ครับ

3. Supply Voltage อยู่ที่ 2.8-3.4 V

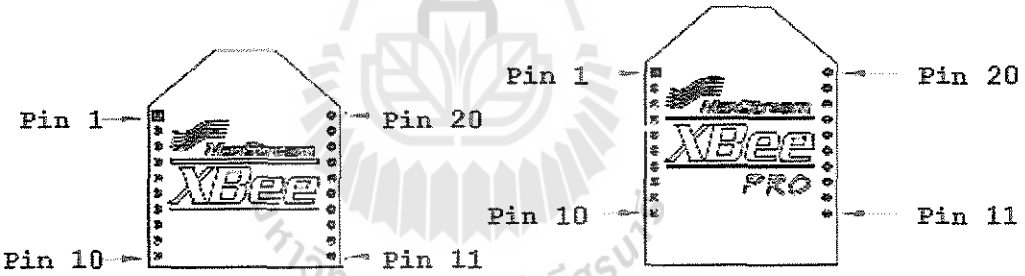
4.Power Down Current< 10uA

5. มี RF data rate อยู่ที่ 250 Kbps (เป็นส่วนของ สัญญาณที่ส่งผ่านอากาศ)

6. มี Serial interface data rate อยู่ระหว่าง 1200 – 115200 Bps (เป็นส่วนที่ติดต่อสื่อสารกับไมโครคอนโทรเลอร์)

7. เป็น Spread Spectrum ชนิด DSSS (Direst Sequence Spread Spectrum)

8. การกำหนด addressing มีลำดับลักษณะคือ กำหนด PAN ID สำหรับเครือข่ายหนึ่ง ๆ , กำหนด Channel และ กำหนด addressของแต่ละตัว



รูปจาก Datasheet ของ Xbee

Xbeeจะมีอยู่ 20 ขา โดยผมของยกตัวอย่างจาก datasheet ที่ ThaiEasyElec ได้มาล่าสุด เป็นของรุ่น Xbee Series1 (Digiซึ่งเป็นบริษัทผู้ผลิต Xbee มักจะมีความเปลี่ยนแปลงเกี่ยวกับสินค้าอยู่บ่อย ๆ ครับ) อธิบายหน้าที่แต่ละขาตามตาราง Datasheet หน้า 7 Xbee ยังมีขาที่เป็น Digital I/O และ Analog to Digital ขนาด 10 bits อีกด้วยแต่ไม่สามารถนำ Xbee ต่อลง Protoboard ได้โดยตรงเพราะความห่างช่องขาของ Xbee แคบกว่าช่อง Protoboard ท่านจะต้องทำ PCB ขึ้นมาเพื่อต่อใช้งานเลย , หรือไม่ก็หา Socket แปลงขาใช้งาน เป็นระยะห่างขามาตรฐาน (Pitch) เท่ากับ 2.54 mmหรือ 1 mil สำหรับผู้ที่ต้องการจะออกแบบ PCB ท่านสามารถดูแบบ Drawing ได้จาก เอกสารนี้

Xbee จะมีอยู่ 2 รุ่นคือ รุ่น series 1 และ รุ่น series 2 (ZB) และยังมีขนาด power ให้เลือกอีก 2 แบบ คือ แบบธรรมดา (1 mw – 2 mw) และ แบบ PRO (50mw- 60 mw) ซึ่งจะมีผลเรื่องระยะทางการรับส่งข้อมูล โดยแต่ละ series นั้น สามารถสร้างเครือข่ายได้หลายแบบ แต่จะมีเพียง series 2 เท่านั้นที่จะทำเครือข่ายแบบ mesh ได้ ซึ่งยังมีรายละเอียดปลีกย่อยในเรื่องของความแตกต่างในแต่ละ series กรุณาดู spec ได้จาก datasheet ของแต่ละ series เพิ่มเติมครับ

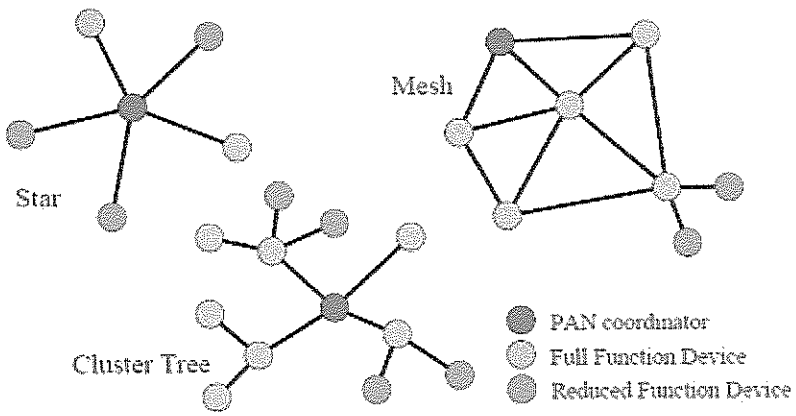
<u>Xbeeทั้ง 2 Series นี้สามารถสร้าง Topologyได้ดังนี้</u>

- รุ่น series 1 Peer-to-peer, point-to-point, point-to-multipoint (Broadcast)
- รุ่น series 2 (ZB) Mesh, Peer-to-peer, point-to-point, point-to-multipoint (Broadcast)

Xbee series2 จะมี parameter ในการ setting ต่าง ๆ มากกว่า series 1

หมายเหตุ

- peer-to-peer network หมายถึง เครือข่ายที่อยู่ในระดับชั้นเดียวกัน ยกตัวอย่าง ใน OSI Layer เช่น ระดับ Transport Network Layer กับ Transport Network Layer นั่นก็คือ TCP Protocol ระหว่างคอมพิวเตอร์ 2 เครื่อง แต่สำหรับ Xbeeคำว่า peer-to-peer network หรือ Non Beacon Network คือ การที่เรา set node เป็น End Device หมดทุกตัว ไม่มีการกำหนดตายตัวว่า ตัวใดจะเป็น Master ตัวใดจะเป็น Slave แต่จะให้ระบบจัดการกันเอง โดยในเครือข่าย จะต้องกำหนด parameter ID (PAN ID) และ CH (Channel)
- Xbeeสามารถ Set ให้เป็น End-Device , Router , Coordinator ตามเงื่อนไขของ Zigbee



เครือข่าย Zigbee แบบ Star , Cluster , Mesh

การศึกษาการใช้งาน Xbeeในเบื้องต้น

เราสามารถทดสอบด้วยการ ปรับ parameter ที่สำคัญต่าง ๆ ผ่าน software user interface ได้ ท่านสามารถ download software user interface ที่ใช้ร่วมกันกับ Xbeeชื่อ X-CTU มาได้ฟรีจาก Digi ครับ การใช้งาน ท่านสามารถอ่านจากคู่มือ X-CTU Configuration & Test Utility Software User Guide

นอกจากใช้ software แล้ว ต้องมีอุปกรณ์ที่จะเชื่อมต่อ Xbee เข้ากับคอมพิวเตอร์เพื่อทำการติดต่อสื่อสารกับ X-CTU ด้วยครับ อุปกรณ์ที่ว่านี้ คือตัวที่จะนำขาบางขาของ Xbee มาต่อเข้ากับ max232 เพื่อเปลี่ยนระดับสัญญาณ TTL ให้สามารถติดต่อสื่อสารกับคอมพิวเตอร์ผ่าน RS232 (DB9) ได้ หรือเราจะใช้ FT232RL สำหรับแปลง serial เป็น USB ในกรณีที่ computer ไม่มีพอร์ต DB9แล้วก็ได้ครับ

ตัวอย่างอุปกรณ์ที่ใช้เชื่อมต่อกับคอมพิวเตอร์เพื่อ Update , Config Parameter โปรแกรม firmware ใหม่ และสำหรับใช้ทำการทดสอบเบื้องต้น เช่น Xbee Socket, Xbee USB Dongle , Xbee Breadboard , X-bee RS232 (DB9) Dongle อุปกรณ์เหล่านี้ถือว่าเป็นอุปกรณ์เสริมที่ลดภาระงานทางด้าน Hardware ได้ส่วนหนึ่ง ในระยะเริ่มต้นก็จะสะดวกดีครับ แต่หากต้องการจะทำ Hardware ขึ้นเอง ก็สามารถดูวงจร Schematic จาก Digi ครับ

กำลังส่ง สายอากาศ และ สัญญาณ รบกวน ของ Xbee

Xbee นั้น ใช้ย่านความถี่ 2.4 Ghz ซึ่งเป็นย่านเดียวกันกับ Bluetooth หรือ Wireless Lan ดังนั้นหลายคนอาจจะสงสัยว่า สัญญาณมันจะกวนกันหรือไม่ คำตอบคือ กวนกันครับ ยกตัวอย่างเช่น ที่ร้าน ThaiEasyElec เรามี wireless lan ใช้ Access Point 1 ตัว และเราก็ยังใช้เครื่องโทรศัพท์แบบ ถือไปมาได้ (2.4 GHz cordless telephones) ซึ่งก็ใช้ย่าน 2.4 Ghz เช่นกัน แล้วเราก็ทดสอบใช้งาน Xbee ด้วยครับ ผลปรากฏว่า การรับส่งสัญญาณก็ขาดหายไปบ้าง แต่บางครั้งก็ครบถ้วน นั่นอาจเป็นเพราะกำลังส่งของ Xbee และระยะทางของ node ที่เราทดสอบใกล้กันมาก

ช่องสัญญาณย่านความถี่ 2.4 Ghz นี้เรียกว่าเป็นย่านไมโครเวฟ หลักสำคัญของย่านไมโครเวฟอย่างหนึ่งคือ การวางตำแหน่งตัวรับส่งสัญญาณนั้น ต้องตั้งแบบ line of sight (ไม่มีสิ่งกีดขวางใด ๆ) ถึงจะได้กำลังส่งสูงสุด สำหรับกำลังส่งของ Xbee ในรุ่น Pro จะใช้ 50-60 mw ใน datasheet บอกว่าได้ไกลถึง 1.5 km. แต่ก็ต้องเป็นลักษณะ ของ line of sight ครับ หากไม่ใช่เงื่อนไขนี้ เราจะได้ระยะการรับส่งสัญญาณที่ลดลง นอกเหนือจากเรื่อง line of sight แล้ว ยังมีเรื่องสัญญาณรบกวนต่าง ๆ

(Interference) มาถึงเรื่องของสายอากาศตัว Xbee มีให้เลือกที่เป็นแบบสำเร็จรูปพร้อมใช้ ไม่ต้องหาสายอากาศมาต่อเพิ่มคือ สายอากาศแบบchip ant และ whip ant ซึ่ง Pattern การแพร่กระจายคลื่นจะบ่งบอกเราอย่างหนึ่ง(จากหลาย ๆ อย่าง) ได้ว่า สายอากาศนี้ จะมีอัตราการขยายดีอย่างไร ท่านสามารถดูเอกสารจาก Digi เพิ่มเติม

Chip ant นั้น ก็มีข้อดีตรงที่มันทำให้ขนาด Dimension รวมมันเล็กลง แต่ Gain น้อยกว่าแบบ Whip ant ครับ Chip ant จึงมีระยะรับส่งข้อมูลที่ลดลงจาก spec ใน datasheet ยกตัวอย่างเช่น รุ่น Pro ที่บอกว่าสามารถส่งได้ไกลสูงสุด 1.5 km แบบ line of sightแต่ถ้าเราเลือก chip ant แล้ว จะได้ระยะสูงสุดอยู่ที่ 500 กว่าเมตร

Module	Antenna Type	Outdoor Distance (Visual Line-of-Sight)	Indoor Distance (Office Building)	Indoor Distance (Warehouse)
XBee	Chip	470 ft. (143 m)	80 ft. (24 m)	-
	Whip	845 ft. (258 m)	80 ft. (24 m)	84 ft. (26 m)
XBee-PRO	Chip	1650 ft. (515 m)	140 ft. (43 m)	-
	Whip	4382 ft. (1335 m)	140 ft. (43 m)	355 ft. (108 m)

Xbee Association

ในเครือข่าย Zigbee ต้องมีการทำงานในโหมดประหยัดพลังงาน ในช่วงเวลาที่ไม่มีการทำงานรับส่งข้อมูล ดังนั้นตัว Xbeeจึงมี Parameter ที่จะกำหนดการทำงานสำหรับ Sleep mode (Parameter A1,A2,SP,ST)

Xbee Addressing

ตัว Xbee จะสามารถกำหนดค่าประจำตัวอ้างอิงของมัน (Address) 2 แบบ คือ แบบ 16 bit address และ 64 bit addressปกติแล้ว Xbee ทุกตัวจะถูกกำหนดค่ามาจากโรงงานเป็น Address 64 bit อยู่แล้ว ซึ่วจะสามารถอ่านค่าได้จาก parameter SH+SLการใช้งาน Address 64 bit สามารถทำได้โดยกำหนด parameter MY ให้มีค่า 0xFFFF หรือ 0xFFFE ส่วน การกำหนด 16 bit addressนั้นทำได้โดย กำหนด parameter MY ให้มีค่าน้อยกว่า 0xFFFE โดยจะเรียกเป็น mode การทำงาน 2 ประเภท คือ

1. Unicast Mode คือ การรับส่งข้อมูล โดยอาศัยหลักการ Acknowledgement คือหากทางด้านส่งนั้นส่งข้อมูลไป แต่ไม่ได้รับAck ตอบกลับจากตัวรับ ก็จะทำการส่งข้อมูลใหม่
2. Broadcast Mode คือการส่งข้อมูลไปยังปลายทางให้ได้รับข้อมูลทุกตัว

มหาวิทยาลัยเทคโนโลยีสุรนารี

Xbee Operation Mode

Xbee จะสามารถแบ่งช่วงการทำงานได้เป็น 5 แบบ คือ

1.Idle Mode โหมดนี้ จะเป็นโหมดที่ไม่ได้รับส่งข้อมูล ตัว Xbee เตรียมที่จะทำงานในโหมดอื่น ๆ ต่อไปทันที หากมีเงื่อนไขบางอย่าง

2.-3. Transmit / Receive Mode (พูดรวม 2 Mode ครับ) คือช่วงที่ Xbee มีการรับ หรือ ส่งข้อมูล โดยจะแบ่งลักษณะการทำงานย่อยออกเป็น Direct กับแบบ Indirect , การกำหนด Address ต้นทางและปลายทาง , Clear Channel Assessment และ การตอบรับ Acknowledgement

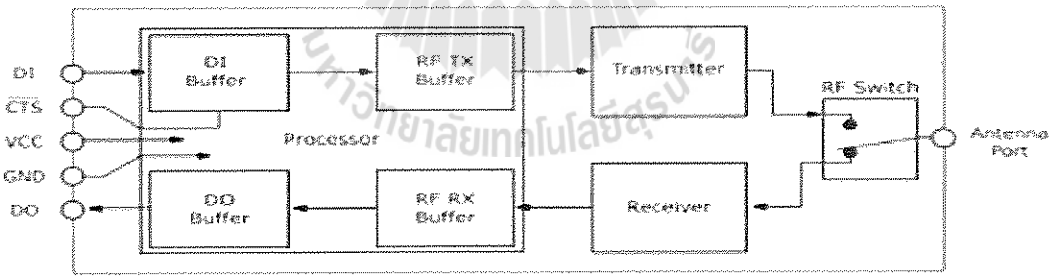
4. Sleep Mode คือ ช่วงที่ Xbee อยู่ในสถานการณ์ทำงานพลังงานต่ำที่สุด เมื่อไม่มีการใช้งาน

5. Command Mode คือ เป็นส่วนการปรับ parameter ของ Xbee ซึ่งจะมีการกำหนด 2 แบบคือ แบบ AT command กับแบบ API Command

โดยบทความนี้จะขอกล่าวถึงเฉพาะ AT Command Mode และกล่าวถึงบาง Parameter ที่เป็นพื้นฐานสำหรับการทำเครือข่ายแบบPoint-to-Point ครับ สำหรับ API command จะขอกล่าวในบทความ Xbee ตอนต่อไปครับ

Data Throughput ของ Xbee

Figure 2-03. Internal Data Flow Diagram



โดยทั่วไปการใช้งาน RF Module ควรจะกำหนดให้มี Buffer ด้วย เพื่อการปรับอัตรารับส่งข้อมูลระหว่างตอนที่รับส่งทางอากาศ กับตอนที่รับส่งไปยัง ไมโครคอนโทรลเลอร์ หรือ อุปกรณ์อื่น ๆ ได้อย่างเหมาะสม

Data ที่รับส่งระหว่าง MCU กับ Xbee จะมีข้อจำกัด เรื่อง Packet อาจถูก Drop ได้ เนื่องมากจาก Data Over Flow โดยสำหรับด้านการส่งข้อมูลไปที่ Xbee เพื่อออกอากาศนั้น ที่ ขา DI จะมี Buffer อยู่ประมาณ 202 Bytes หากส่งเกิน Buffer จะเกิดการ Drop packet ทิ้ง ซึ่งทางฝั่งรับข้อมูลที่ขา DO ก็มี Buffer อยู่เช่นกัน โดยจะมี Parameter ที่เกี่ยวข้องกับ Data Throughput คือ ROและ BD

ค่า RO คือค่า Packetization Timeout ซึ่งเป็น delay ของข้อมูลที่อยู่ใน DI Buffer ก่อนที่จะถูก encapsulate ไปที่ส่วน RF transmission เพื่อส่งข้อมูลออกอากาศ หากตั้ง RO = 0 Data ที่รับเข้ามาจาก MCU จะถูก Xbee Encapsulate Packet ส่งออกอากาศทันที ดังนั้นเราจะมี Parameter RO และ BD ที่จะช่วยในการปรับ Data รับส่งให้สามารถรับส่งกันได้ทัน ไม่ให้มีการ Drop Packet ได้ในกรณีที่ส่งข้อมูลเกิน 200 Bytes

นอกจากนี้ยังมี PIN CTS(ขา12) และ RTS(ขา16) ช่วยเตือนเราเวลาที่ Buffer ภายในใกล้จะเต็มด้วยครับ โดยในฝั่งส่ง DI Buffer จะส่ง Signal มาทาง CTS เมื่อ DI Buffer เหลือพื้นที่จัดเก็บอยู่อีก 17 Bytes และส่ง Clear Signal ที่ CTS เมื่อ DI Buffer เหลือพื้นที่จัดเก็บมากกว่า 34 Bytes

ดังนั้น สำหรับการเขียนโปรแกรม รับส่งข้อมูลกับ Xbee ต้องคำนึงเรื่อง Buffer ด้วยครับ แต่ในทางปฏิบัติ สำหรับงาน Sensor Network ก็ไม่ได้รับส่งข้อมูล Stream Data ยาว ๆ

มหาวิทยาลัยเทคโนโลยีสุรนารี

ภาคผนวก ข

1.Code Program

ส่วนที่ 1 main.java

```
package comport;
importgnu.io.CommPortIdentifier;
importgnu.io.NoSuchPortException;
importjava.awt.Color;
importjava.awt.Toolkit;
importjava.awt.event.ActionEvent;
importjava.awt.event.ActionListener;
importjavax.swing.JOptionPane;
importjavax.swing.Timer;

public class main extends javax.swing.JFrame implements ActionListener{
    Timer t = new Timer(1,this);
private static String stageconnect="";
   private static String s1="",s2="",s3="";
public main() {
t.start();
initComponents();
printliscom();
setIcon();
p1.setBackground(Color.GREEN);
p2.setBackground(Color.GREEN);
p3.setBackground(Color.GREEN);
   }
```

ประกาศตัวแปรเพื่อเก็บค่าต่างๆ ที่ถูกส่งมาจาก arduino

Timer สำหรับ ตั้งเวลา automatic reset ส่วนติดต่อผู้ใช้งาน

เริ่มต้นโปรแกรม อ่านค่าว่ามี Port อะไรเปิดบ้าง และ set ค่า bed1 bed2 bed3 ให้เป็นสีเขียว

```
public void showerror(Exception e){
JOptionPane.showMessageDialog(this,e,"Error",JOptionPane.ERROR_MESSAGE);
   }
public void showmessage(String message){
JOptionPane.showMessageDialog(this,message,"Message",JOptionPane.PLAIN_MESSA
GE);
   }
public void setstatepro(String message){
main.stageconnect = message;
   }
public void setp1(String bed1){
      main.s1 = bed1;
   }
public void setp2(String bed2){
      main.s2 = bed2;
   }
public void setp3(String bed3){
      main.s3 = bed3;
   }
```

```
@Override
public void actionPerformed(ActionEvent e){
if(s1.equals("S11")){
p1.setBackground(Color.RED);
      }
if(s1.equals("S10")){
p1.setBackground(Color.GREEN);
      }
if(s2.equals("S21")){
p2.setBackground(Color.RED);
      }
if(s2.equals("S20")){
p2.setBackground(Color.GREEN);
      }
if(s3.equals("S31")){
p3.setBackground(Color.RED);
      }
if(s3.equals("S30")){
p3.setBackground(Color.GREEN);
      }
else{


      }
  }
```

Method actionPerformedเป็น method ที่ถูก Override มาจาก ActionListener ผลก็คือ ตัว ตัว timer ที่ถูกประกาศไว้ตั้งแต่ต้น จะทำการ เขียน หน้า ติดต่อผู้ใช้ใหม่ ทุก ๆ ครั้งที่เกิดการเปลี่ยนแปลง ค่าที่อยู่ในโปรแกรม ส่วนนี้ คือส่วนที่ เขียนหน้าผู้ใช้ใหม่ เช่น ถ้าค่าที่ถูกส่งมาจาก arduinoเป็น S11 ก็จะแสดงเป็น สีแดงที่ Bed1 แต่ถ้า เป็น S10 ก็จะแสดงสีเขียวที่ Bed1 เป็นต้น

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
@SuppressWarnings("unchecked")
   // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-
BEGIN:initComponents
private void initComponents() {

selectcom = new javax.swing.JComboBox();
comlabel = new javax.swing.JLabel();
connect = new javax.swing.JButton();
close = new javax.swing.JButton();
      jLabel2 = new javax.swing.JLabel();
statepro = new javax.swing.JLabel();
      p2 = new javax.swing.JPanel();
      p1 = new javax.swing.JPanel();
      bed1 = new javax.swing.JLabel();
      bed2 = new javax.swing.JLabel();
      p3 = new javax.swing.JPanel();
      bed3 = new javax.swing.JLabel();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
setTitle("Baby Soft V1.0");
setBounds(new java.awt.Rectangle(300, 250, 0, 0));
setMaximizedBounds(new java.awt.Rectangle(300, 250, 0, 0));
setModalExclusionType(java.awt.Dialog.ModalExclusionType.APPLICATION_EXCLU
DE);
setName("desktop"); // NOI18N
setType(java.awt.Window.Type.POPUP);
```

```
selectcom.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
selectcom.addActionListener(new java.awt.event.ActionListener() {
public void actionPerformed(java.awt.event.ActionEventevt) {
selectcomActionPerformed(evt);
        }
    });

comlabel.setFont(new java.awt.Font("JasmineUPC", 1, 24)); // NOI18N
comlabel.setText("COM PORT");

connect.setFont(new java.awt.Font("JasmineUPC", 1, 24)); // NOI18N
connect.setText("CONNECT");
connect.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
connect.addMouseListener(new java.awt.event.MouseAdapter() {
public void mouseClicked(java.awt.event.MouseEventevt) {
connectMouseClicked(evt);
        }
    });

close.setFont(new java.awt.Font("JasmineUPC", 1, 24)); // NOI18N
close.setText("CLOSE");
close.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
close.addMouseListener(new java.awt.event.MouseAdapter() {
public void mouseClicked(java.awt.event.MouseEventevt) {
closeMouseClicked(evt);
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        }
    });

jLabel2.setText("สถานะการเชื่อมต่อ");

statepro.setForeground(new java.awt.Color(255, 51, 51));
statepro.setText("OFFLINE");

p2.setBackground(new java.awt.Color(255, 204, 0));
p2.setForeground(new java.awt.Color(0, 255, 0));
p2.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
p2.setFocusCycleRoot(true);
p2.setFocusTraversalPolicyProvider(true);

javax.swing.GroupLayout p2Layout = new javax.swing.GroupLayout(p2);
p2.setLayout(p2Layout);
p2Layout.setHorizontalGroup(
p2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGap(0, 80, Short.MAX_VALUE)
    );
p2Layout.setVerticalGroup(
p2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGap(0, 70, Short.MAX_VALUE)
    );

p1.setBackground(new java.awt.Color(255, 204, 0));
```

```
p1.setForeground(new java.awt.Color(0, 255, 0));
p1.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));

javax.swing.GroupLayout p1Layout = new javax.swing.GroupLayout(p1);
p1.setLayout(p1Layout);
p1Layout.setHorizontalGroup(
p1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGap(0, 80, Short.MAX_VALUE)
    );
p1Layout.setVerticalGroup(
p1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGap(0, 70, Short.MAX_VALUE)
    );

bed1.setFont(new java.awt.Font("JasmineUPC", 1, 24)); // NOI18N
bed1.setText("BED 1");

bed2.setFont(new java.awt.Font("JasmineUPC", 1, 24)); // NOI18N
bed2.setText("BED 2");

p3.setBackground(new java.awt.Color(255, 204, 0));
p3.setForeground(new java.awt.Color(0, 255, 0));
p3.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));

javax.swing.GroupLayout p3Layout = new javax.swing.GroupLayout(p3);
p3.setLayout(p3Layout);
```

```
p3Layout.setHorizontalGroup(
p3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGap(0, 80, Short.MAX_VALUE)
    );
p3Layout.setVerticalGroup(
p3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGap(0, 70, Short.MAX_VALUE)
    );

bed3.setFont(new java.awt.Font("JasmineUPC", 1, 24)); // NOI18N
bed3.setText("BED 3");

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
                .addGap(160, 160, 160)
                .addComponent(jLabel2)
                .addGap(18, 18, 18)
                .addComponent(statepro, javax.swing.GroupLayout.PREFERRED_SIZE,
58, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGroup(layout.createSequentialGroup()
```

```
                .addGap(242, 242, 242)
                .addComponent(p3, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGroup(layout.createSequentialGroup()
                .addGap(29, 29, 29)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                    .addComponent(bed3)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING,
false)
                        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
                            .addComponent(connect)
                            .addGap(18, 18, 18)
                            .addComponent(close)
                            .addGap(25, 25, 25))
                        .addGroup(layout.createSequentialGroup()
                            .addComponent(comlabel)
                            .addGap(18, 18, 18)
                            .addComponent(selectcom,
javax.swing.GroupLayout.PREFERRED_SIZE, 169,
javax.swing.GroupLayout.PREFERRED_SIZE))))))
            .addGroup(layout.createSequentialGroup()
                .addGap(20, 20, 20)
```

```
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(p1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGroup(layout.createSequentialGroup()
                  .addGap(17, 17, 17)
                  .addComponent(bed1)))
              .addGap(28, 28, 28)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                  .addComponent(p2, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                  .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
                .addGroup(layout.createSequentialGroup()
                  .addGap(15, 15, 15)
                  .addComponent(bed2)
                  .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))))
      );
layout.setVerticalGroup(
layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
```

```
                .addContainerGap()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(selectcom, javax.swing.GroupLayout.PREFERRED_SIZE,
29, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(comlabel))
                .addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(connect)
                    .addComponent(close))

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addGroup(layout.createSequentialGroup()
                        .addGap(44, 44, 44)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                            .addComponent(bed2,
javax.swing.GroupLayout.Alignment.TRAILING)
                            .addComponent(bed1,
javax.swing.GroupLayout.Alignment.TRAILING)))
                    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                        .addComponent(bed3)))
```

```
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                .addComponent(p3, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(p2, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(p1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
38, Short.MAX_VALUE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel2)
                .addComponent(statepro, javax.swing.GroupLayout.PREFERRED_SIZE,
14, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addContainerGap())
        );

p2.getAccessibleContext().setAccessibleParent(p2);

pack();
    }// </editor-fold>//GEN-END:initComponents
```

ในหน้า ถัดไปคือส่วนการทำงานเมื่อ มีการคลิกปุ่ม connect โดยเมื่อเรากดปุ่ม connect มันจะไปเรียก SerialTest main = new SerialTest();เก็บไว้ในตัวแปร main โดย ไฟล์ SerialTest.java จะทำหน้าที่ ไปพร้อมๆ กับ ตัว main.java สิ่งนี้เรียกว่าการ Thread ซึ่งเป็นการทำงานแบบ คู่ขนาน ผลก็คือ เมื่อ เกิดการส่งค่า มาจาก arduinoตัว โดยส่งมาจาก SerialTest.java ตัว main.java ก็จะเปลี่ยนแปลงไปด้วย และ Timer จะทำการ reset ค่าที่เปลี่ยนไป เป็นผลปรากฏให้ เห็น สีของเตียงนั้นเปลี่ยนไป ซึ่งรูปแบบ การทำงาน แบบ Thread จะส่งผลให้ตัว โปรแกรมมีผลต่อการตอบสนองกับ สัญญาณที่มา จาก arduinoได้เร็วขึ้น แต่ก็เป็น ผลเสีย ต่อการใช้ ทรัพยากรณ์ในคอมพิวเตอร์ บางชนิด ซึ่ง จะทำให้เกิดอาการค้างใน คอมรุ่น ที่ไม่รองรับ Thread ได้

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
private void connectMouseClicked(java.awt.event.MouseEventevt) {//GEN-
FIRST:event_connectMouseClicked


        SerialTest main = new SerialTest();
        main.initialize();
                Thread t=new Thread() {
                    @Override
        public void run() {
                        //the following line will keep this app alive for 1000 seconds,
        //waiting for events to occur and responding to them (printing incoming messages to
        console).
        try {
        Thread.sleep(1000);
        statepro.setText(stageconnect);
        selectcom.disable();
                    }
        catch (InterruptedExceptionie) {
        showerror(ie);
                    }
                }
            };
        t.start();
           }//GEN-LAST:event_connectMouseClicked
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

ส่วน การทำงานเมื่อ คลิกปุ่ม connect

```
private void closeMouseClicked(java.awt.event.MouseEventevt) {//GEN-
FIRST:event_closeMouseClicked
System.exit(0);
    }//GEN-LAST:event_closeMouseClicked
```

เมื่อกดปุ่ม CLOSE จะเป็นการปิดโปรแกรมด้วยคำสั่ง System.exit(0)

```
private void selectcomActionPerformed(java.awt.event.ActionEventevt) {//GEN-
FIRST:event_selectcomActionPerformed
SerialTest main = new SerialTest();
main.setportname(selectcom.getSelectedItem().toString());
    }//GEN-LAST:event_selectcomActionPerformed
```

เมื่อเราคลิกเลือก comportที่ต้องการจะเชื่อมต่อ จะเป็นการ set ค่าให้กับ setportnameเพื่อ ระบุว่า comportที่จะเชื่อมต่อคือ port อะไร เพื่อนำไปใช้ในการเชื่อมต่อ port นั้นต่อไป

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
public static void main(String args[]) {
java.awt.EventQueue.invokeLater(new Runnable() {
        @Override
public void run() {
new main().setVisible(true);
        }
    });
  }
```

```
public final void printliscom(){
for(int i=0;i<15;i++){
        String comPort = "COM"+i;
try{
CommPortIdentifierportId = CommPortIdentifier.getPortIdentifier(comPort);
selectcom.addItem(portId.getName());
        }
catch(NoSuchPortException e){

        }
    }
  }
```

Method นี้คือส่วนที่จะแสดง ออกมาทาง โปรแกรมว่ามี comportใดๆ บ้างที่ถูกใช้งานในคอมพิวเตอร์เครื่องนี้อยู่ เพื่อให้ผู้ใช้ได้เลือก comport ที่ต้องการ connect

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
    // Variables declaration - do not modify//GEN-BEGIN:variables
privatejavax.swing.JLabel bed1;
privatejavax.swing.JLabel bed2;
privatejavax.swing.JLabel bed3;
privatejavax.swing.JButton close;
privatejavax.swing.JLabelcomlabel;
privatejavax.swing.JButton connect;
privatejavax.swing.JLabel jLabel2;
privatejavax.swing.JPanel p1;
privatejavax.swing.JPanel p2;
privatejavax.swing.JPanel p3;
privatejavax.swing.JComboBoxselectcom;
privatejavax.swing.JLabelstatepro;
    // End of variables declaration//GEN-END:variables


private void setIcon() {
setIconImage(Toolkit.getDefaultToolkit().getImage(getClass().getResource("/img/baby.j
pg")));
    }
}
```

ส่วนที่ 2 SerialTest.java

```
package comport;

importjava.io.BufferedReader;

importjava.io.InputStreamReader;

importjava.io.OutputStream;

importgnu.io.CommPortIdentifier;

importgnu.io.PortInUseException;

importgnu.io.SerialPort;

importgnu.io.SerialPortEvent;

importgnu.io.SerialPortEventListener;

importgnu.io.UnsupportedCommOperationException;

importjava.io.IOException;

importjava.util.Enumeration;

importjava.util.TooManyListenersException;


public class SerialTest implements SerialPortEventListener {

        SerialPortserialPort;

    /** The port we're normally going to use. */

        public static final String PORT_NAMES[] = {

                        "/dev/tty.usbserial-A9007UX1", // Mac OS X

                        "/dev/ttyUSB0", // Linux

                        "COM3", // Windows

        };
```

```
public void setportname(String comport){
        PORT_NAMES[2] = comport;
    }
```

ส่วนการตั้งค่า Comport ที่เกิดจากการเลือกของผู้ใช้ว่าจะเชื่อม Comport ไหน และถูกส่งมาจาก main.java

```
    /**
    * A BufferedReader which will be fed by aInputStreamReader
    * converting the bytes into characters
    * making the displayed results codepage independent
    */
privateBufferedReader input;
/** The output stream to the port */
privateOutputStream output;
/** Milliseconds to block while waiting for port open */
private static final int TIME_OUT = 2000;
/** Default bits per second for COM port. */
private static final int DATA_RATE = 9600;
```

เป็นการประกาศตัวแปรเพื่อระบุรูปแบบการเชื่อมต่อ ของ Computer กับ arduino

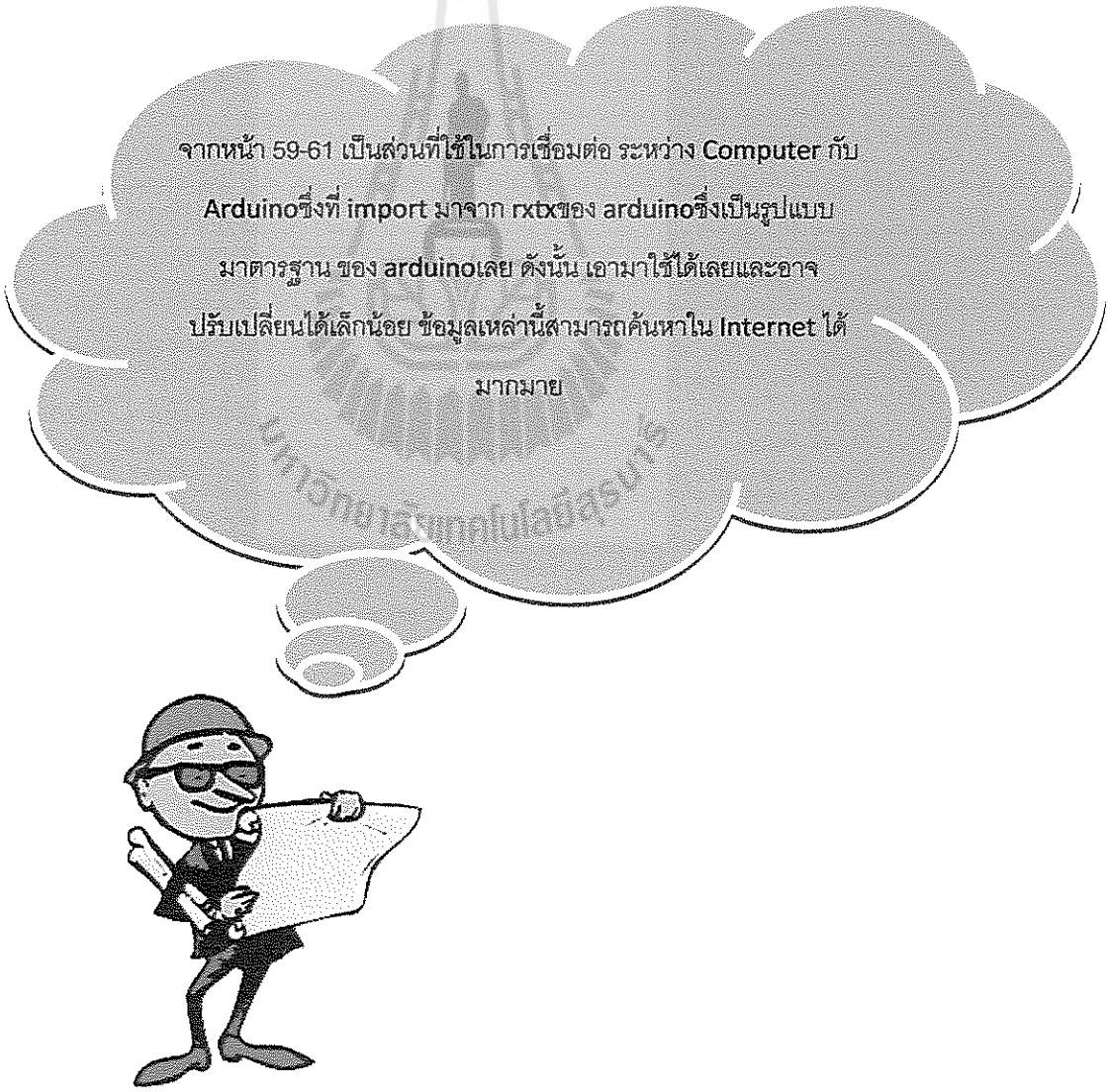
```
public void initialize() {
        CommPortIdentifierportId = null;
        Enumeration portEnum = CommPortIdentifier.getPortIdentifiers();

        //First, Find an instance of serial port as set in PORT_NAMES.
        while (portEnum.hasMoreElements()) {

CommPortIdentifiercurrPortId = (CommPortIdentifier) portEnum.nextElement();
                for (String portName : PORT_NAMES) {
                        if (currPortId.getName().equals(portName)) {
new main().showmessage("Start "+portName);
new main().setstatepro("ONLINE");
                                portId = currPortId;
                                break;
                        }
                }
        }
        if (portId == null) {
new main().showmessage("Could not find COM port.");
                return;
        }

        try {
// open serial port, and use class name for the appName.
serialPort = (SerialPort) portId.open(this.getClass().getName(),
TIME_OUT);
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
// set port parameters
serialPort.setSerialPortParams(DATA_RATE,
        SerialPort.DATABITS_8,
        SerialPort.STOPBITS_1,
        SerialPort.PARITY_NONE);

// open the streams
input = new BufferedReader(newInputStreamReader(serialPort.getInputStream()));
output = serialPort.getOutputStream();

// add event listeners
        serialPort.addEventListener(this);
        serialPort.notifyOnDataAvailable(true);
} catch (PortInUseException | UnsupportedCommOperationException | IOException |
TooManyListenersException e) {
        new main().showerror(e);
                }
        }

        /**
         * This should be called when you stop using the port.
         * This will prevent port locking on platforms like Linux.
         */
        public synchronized void close() {
                if (serialPort != null) {
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
                serialPort.removeEventListener();
                serialPort.close();
            }
        }

        /**
         * Handle an event on the serial port. Read the data and print it.
* @paramoEvent
         */
```



```
@Override
    public synchronized void serialEvent(SerialPortEventoEvent) {
            if (oEvent.getEventType() == SerialPortEvent.DATA_AVAILABLE) {
                try {
    String inputLine=input.readLine();
if(inputLine.equals("S11")){
new main().setp1(inputLine);
                }
if(inputLine.equals("S10")){
new main().setp1(inputLine);
                }
if(inputLine.equals("S21")){
new main().setp2(inputLine);
                }
if(inputLine.equals("S20")){
new main().setp2(inputLine);
                }
if(inputLine.equals("S31")){
new main().setp3(inputLine);
                }
if(inputLine.equals("S30")){
new main().setp3(inputLine);
                }
                } catch (IOException e) {
                    //new main().showerror(e);
                }
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
            }
            // Ignore all the other eventTypes, but you should consider the other ones.
        }
    }
```

จากหน้า 62-63 เป็นการรับค่ามาจาก Arduinoและก็ Process เพื่อส่งค่ากลับไปให้ main.java และ แสดงผลออกทางคอมพิวเตอร์อีกที การทำงาน ก็ คือ ตัว Arduinoจะส่งค่า ปกติไม่เกิดความชื้นมาเป็น S10 S20 S30 ตามลำดับ เมื่อเป็นเช่นนั้น ก็จะส่งไปให้ main.java และ main.java ก็จะแสดงผล ออกมาเป็นสีเขียว แต่ถ้า Arduinoส่ง S11 S21 S31 ก็จะส่งผลให้ main.java แสดงค่าเป็นสีแดงด้วย สรุปคือส่วนนี้ คือส่วนที่อ่านค่าจาก Arduinoที่ส่งมาให้ Computer ผ่าน Comport นั่นเอง

ส่วนที่ 3 ContorlAduinoBord

```
int pushButton_1=2;
int pushButton_2=3;
void setup(){
Serial.begin(9600);
pinMode(pushButton_1,INPUT);
}
void loop(){
int buttonState_1=digitalRead(pushButton_1);
if (buttonState_1 == 0)
 {      Serial.println("S11");
        delay(1000);
 }
if (buttonState_1 == 1)
 {
        Serial.println("S10");
        delay(1000);
 }
}
```

Serial.println(“S11”);

เลข1ลำดับที่1: กำหนดหมายเลขอุปกรณ์

เลข1ลำดับที่2: กำหนดลอจิกที่ต้องการ

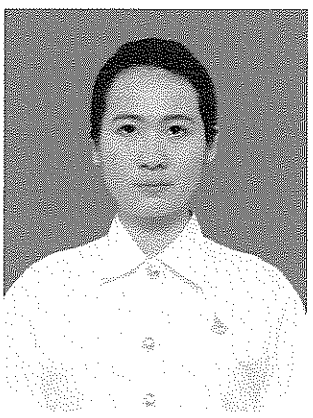
**(1=ชื้น),(0=ไม่ชื้น)

มหาวิทยาลัยเทคโนโลยีสุรนารี

ประวัติผู้เขียน

นางสาวปานหทัย แซ่เตี๋ย เกิดเมื่อวันที่ 5 พฤศจิกายน พ.ศ. 2534 ภูมิลำเนาอยู่ที่ ตำบลหนองบัว อำเภอเมือง จังหวัดหนองบัวลำภู สำเร็จการศึกษาระดับมัธยมศึกษาตอนต้น-ตอนปลายจากโรงเรียนสุรนารีวิทยา 2 อำเภอเมือง จังหวัดนครราชสีมา เมื่อปี พ.ศ. 2553 ปัจจุบันเป็นนักศึกษาชั้นปีที่ 4 สาขาวิชาวิศวกรรมโทรคมนาคม สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

นางสาวณัฐฐา คลังสินธ์ เกิดเมื่อวันที่ 22 มกราคม พ.ศ. 2535 ภูมิลำเนาอยู่ที่ ตำบลพนัสนิคม อำเภอพนัสนิคม จังหวัดชลบุรี สำเร็จการศึกษาระดับมัธยมศึกษาตอนต้น-ตอนปลายจากโรงเรียนพนัสพิทยาคาร อำเภอพนัสนิคม จังหวัดชลบุรี เมื่อปี พ.ศ.2553 ปัจจุบันเป็นนักศึกษาชั้นปีที่ 4 สาขาวิชาวิศวกรรมโทรคมนาคม สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี



นางสาวกีรติกานต์ ใจเอื้อ เกิดเมื่อวันที่ 23 พฤศจิกายน พ.ศ. 2534 ภูมิลำเนาอยู่ที่ ตำบลธารปราสาท อำเภอโนนสูง จังหวัดนครราชสีมา สำเร็จการศึกษาระดับมัธยมศึกษาตอนปลายจากโรงเรียนพิมายวิทยา อำเภอพิมาย จังหวัดนครราชสีมา เมื่อปี พ.ศ. 2553 ปัจจุบันเป็นนักศึกษาชั้นปีที่ 4 สาขาวิชาวิศวกรรมโทรคมนาคม สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

มหาวิทยาลัยเทคโนโลยีสุรนารี